



Universidad
Carlos III de Madrid

PROYECTO FIN DE CARRERA

**Desarrollo de una herramienta educativa para el
aprendizaje a través de vídeos y dispositivos
móviles**

INGENIERÍA TÉCNICA EN INFORMÁTICA DE GESTIÓN

Autor: Paulina Silva Simbaña

Tutor: Dr. David Griol Barres

Leganés, enero de 2016

Título: Desarrollo de una herramienta educativa para el aprendizaje a través de vídeos y dispositivos móviles

Autor: David Griol Barres

Tutor: Paulina Silva Simbaña

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día ____ de _____ de 2016 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

Principalmente a mis padres, Magdalena y Fausto, quienes han sido mi primordial y más grande apoyo en este camino, valoro el enorme esfuerzo que les ha supuesto darme la mayor herencia que se puede dejar a un hijo, la Educación.

A mi hermano por sus horas dedicadas ayudándome como él ha podido para culminar este proyecto.

A Alexis y su hermano que son la razón de mi vida.

A Robin mi compañero en las buenas y en las malas.

A mis compañeros Marcos por sus horas dedicadas para brindarme un poco de luz y a Jenny por sus palabras de apoyo.

Finalmente a David por su colaboración y paciencia que ha tenido conmigo para culminar el proyecto de la mejor manera posible.

Resumen

Actualmente se buscan métodos didácticos para ayudar a los estudiantes en el proceso de aprendizaje mediante una forma fácil y divertida, es aquí donde la música y los videos juegan un papel muy importante para conseguir asimilar la lección de la manera más natural posible.

Para el aprendizaje de cualquier tipo de asignatura o de cualquier tema en general se acude principalmente a la búsqueda de Webs que describen por escrito el asunto de interés, o mejor aún directamente a Youtube, donde sin duda encuentras videos de personas que brindan sus conocimientos para ayudar al internauta.

Es precisamente los Videos el objetivo del desarrollo del proyecto, se da la oportunidad al usuario de interactuar con el video contestando a una serie de preguntas que aparecerán de forma automática a lo largo de la reproducción, el propósito de las preguntas es ayudar al propio usuario a comprobar si él video está siendo comprendido por él o no.

El tipo de contestación que se den a las preguntas condicionaran el comportamiento del video, es decir si se contesta correctamente, el video avanzará hasta mostrar la siguiente pregunta, en caso contrario volverá a reproducir desde el segundo después al que apareció la última pregunta contestada de forma correcta.

El video se reproducirá hasta el final siempre y cuando el usuario haya contestado correctamente a todas y cada una de las preguntas, en el caso de contestar de forma equivocada, se irán repitiendo tramos del video tantas veces como contestaciones incorrectas proporcione el usuario, la única forma de avanzar en la reproducción es acertar en la solución, bien porque el usuario ha conseguido comprender cuál es la contestación correcta a través de las explicaciones o de tanto repetir el mismo tramo del video se ha ido probando con cada una de las respuestas disponibles hasta encontrar la exacta.

Los videos deben ser preparados de tal manera que las preguntas aparezcan en los tiempos establecidos y así conseguir una interacción didáctica con sentido, por este motivo es importante indicar a las personas que facilita el video lo trascendente de realizar una correspondencia correcta entre tramo de video reproducido y pregunta realizada en ese mismo tramo.

Para el desarrollo de la aplicación se ha utilizado como lenguaje de programación Groovy, un lenguaje orientado a objetos que se ejecuta sobre la plataforma Java y Grails, un framework para el desarrollo con Groovy.

Abstract

At present they look for didactic methods to help the students in the learning process by means of an easy and entertaining form, it is here where the music and the videos play a very important role to manage to assimilate the lesson of a way as natural as possible.

For the learning of any type of subject or any subject generally it is gone mainly to the search of Webs that describe the interest subject in writing, or better still directly to YouTube, where without a doubt you find videos of people who offer their knowledge to help the net user.

The objective of the development of the project is indeed the Videos, the opportunity occurs to the user to interact with the video answering a series of questions that will appear of automatic form throughout the reproduction, the intention of the questions is to help the own user to verify if it video is being understood by him or no.

The type of answer that occurs to the questions conditioned the behavior of the video, that is to say if it is answered correctly, the video will advance until showing the following question, on the contrary will return to reproduce later from the second to which correctly it appeared the last answered question.

The video will reproduce until the end as long as the user has answered each and every one of the questions correctly, in the case of answering of mistaken form, will be repeated sections of the video so many times as incorrect answers provide the user, the only form to advance in the reproduction is to guess right in the solution, or because the user has been able to understand which is the correct answer through the explanations or of as much repeating the same section of the video it has been tried on with each one of the answers available until finding the exact one.

The videos must be preparations in such a way that the questions appear in the established times and thus to secure a didactic interaction with sense, for this reason are important to indicate the people that facilitates the video the transcendent to realize a correct correspondence enters section of reproduced video and ask realized in that same section.

The application is developed using like computer language Groovy, language faced to objects that is executed on the platform Java and Grails, a framework for the development with Groovy.

Índice General

CAPÍTULO 1.....	13
1. INTRODUCCIÓN Y OBJETIVOS.....	13
1.1 INTRODUCCIÓN	14
1.1.1 Contexto y Motivación	14
1.1.2 Objetivos	15
1.1.3 Estructura de la Memoria	15
1.1.4 Planificación Temporal	16
1.1.5 Recursos	19
1.1.6 Presupuesto	19
CAPÍTULO 2.....	22
2. ESTADO DEL ARTE	22
2.1 IMPORTANCIA DE LA TIC EN LA EDUCACIÓN	23
2.2 MULTIMEDIA COMO HERRAMIENTA EDUCATIVA.....	25
2.3 VIDEO - ELEMENTO MULTIMEDIA PARA LA ENSEÑANZA	27
2.3.1. Video como un factor de influencia en la motivación, la satisfacción al aprendizaje exitoso	28
2.3.2 Videos Educativo.....	31
2.3.2.1 Medio de Observación.....	31
2.3.2.2 Medio de Expresión	32
2.3.2.3 Medio de Autoaprendizaje	32
2.3.2.4 Medio de Ayuda a la Enseñanza	33
2.4 SISTEMA DE APRENDIZAJE MEDIANTE EL USO DE VIDEOS	34
2.4.1 Introducción.....	34
2.4.2 Módulos que conforma el Sistema	34
2.4.2.1 Módulo de Registro de Videos didácticos en la aplicación	34
2.4.2.2 Módulo Consulta en la Base de Datos	35
2.4.2.3 Módulo de Interacción con los Videos.....	35
2.4.2.4 Módulo de Colaboración con la Aplicación.....	36
2.4.3 HTML5 como soporte integrado para el contenido multimedia	36
2.4.3.1 Evolución de HTML	36
2.4.3.2 HTML5	38
2.4.4 Aplicaciones con Videos Educativos	39
2.5 VENTAJAS QUE OFRECEN LOS VIDEOS.....	42
CAPÍTULO 3.....	43
3. TECNOLOGÍA Y HERRAMIENTAS	43
3.1 LENGUAJES DE PROGRAMACIÓN	44
3.1.1 Java	44
3.1.2 Lenguajes Dinámicos	44
3.1.2.1 Groovy	44
3.1.3 Html5.....	45
3.1.4 JavaScript	46
3.2 FRAMEWORK DE DESARROLLO.....	48
3.2.1 Grails	48

3.3 PLUGINS.....	50
3.3.1 jQuery.....	50
3.3.2 jQuery-ui.....	50
3.3.3 Mail	50
3.3.4 Simple Captcha	50
3.4 ENTORNOS DE DESARROLLO INTEGRADOS (IDE)	50
3.4.1 SpringSource tool suite	50
3.5 OTRAS HERRAMIENTAS.....	51
3.5.1 Firebug	51
3.5.2 Video DownloadHelper	52
3.5.3 Gimp 2	53
CAPÍTULO 4.....	54
4. ANÁLISIS DEL SISTEMA.....	54
4.1 IDENTIFICACIÓN DE ESCENARIOS Y USUARIOS.....	55
4.1.1 Usuarios.....	55
4.1.1.1 Clientes	55
4.1.1.2 Administrador	55
4.2 ESTABLECIMIENTOS DE REQUISITOS DEL SOFTWARE	56
4.2.1 Requisitos Funcionales	57
4.2.2 Requisitos No Funcionales	69
4.3 ESPECIFICACIÓN DE CASOS DE USO	73
4.3.1 Administrador del Sistema.....	75
4.3.2 Usuarios Consumidor	77
4.3.3 Usuarios Colaborador	80
CAPÍTULO 5.....	82
5. DISEÑO DE LA APLICACIÓN	82
5.1 ARQUITECTURA DE LA APLICACIÓN	83
5.1.1 Arquitectura Cliente-Servidor.....	83
5.1.2 Modelo Vista Controlador.....	83
5.2 MODELO DE DATOS	85
5.2.1 UML.....	86
5.2.2 Modelo Relacional	86
5.2.2.1 Diagrama UML de la aplicación	86
5.3 VISTAS DE LA APLICACIÓN.....	89
5.3.1 Interfaces Comunes	91
5.3.1.1 Pantalla Principal	91
5.3.1.2 Listado de Búsqueda General	91
5.3.1.3 Listado de Búsqueda por SubAsignaturas.....	92
5.3.1.4 Reproducción de Video.....	93
5.3.1.5 Preguntas del Video.....	93
5.3.2 Interfaz de Usuario Colaborador	95
5.3.3 Interfaz de Administrador	96
5.3.3.1 Pantalla Principal	96
5.3.3.2 Gestionar	97
5.3.3.2.1 Gestionar Asignaturas.....	98
5.3.3.2.2 Gestionar SubAsignaturas	100
5.3.3.2.3 Gestionar Videos.....	103
5.3.3.2.2 Gestionar Preguntas	106

5.4 CONTROLADORES DE LA APLICACIÓN	108
CAPÍTULO 6.....	110
6. IMPLEMENTACIÓN DE LA APLICACIÓN	110
6.1 ESTRUCTURA DE LA APLICACIÓN	111
6.2 CÓDIGO DE LA APLICACIÓN	112
6.2.1 Clases de dominio	112
6.2.2 Controladores	114
6.2.3 Vistas.....	123
6.2.4 Servicios.....	128
6.2.5 JavaScript	129
6.2.6 Conf	130
CAPÍTULO 7.....	132
7. PRUEBAS.....	132
7.1 PRUEBAS REALIZADAS.....	133
7.1.1 Pruebas de Caja Blanca.....	133
7.1.1.1 Pruebas de Búsqueda General.....	133
7.1.1.2 Pruebas con Formulario de Colaboración.....	134
7.1.1.3 Pruebas al interactuar con videos	135
7.1.1.4 Pruebas con Administrador	136
7.1.1.5 Pruebas Gestionar	137
CAPÍTULO 8.....	139
8. CONCLUSIONES Y TRABAJOS FUTUROS	139
8.1 CONCLUSIONES DEL PROYECTO.....	140
8.1.1 Uso de tecnologías ágiles.....	140
8.1.1.1 Html5	140
8.1.1.2 Javascript	140
8.1.1.3 Groovy y Grails.....	141
8.1.2 Pruebas.....	142
8.1.3 Conclusión Personal	142
8.2 TRABAJOS FUTUROS.....	143
CAPÍTULO 9.....	146
9. BIOGRAFÍA Y REFERENCIAS	146

Índice de Figuras

FIGURA 1. DIAGRAMA DE GANTT	18
FIGURA 2. TEORÍA COGNITIVA DEL APRENDIZAJE MULTIMEDIA (MAYER, 2001)	28
FIGURA 3. ICONO DE HTML	36
FIGURA 4. HTML5	38
FIGURA 5. RETENCIÓN DE CONTENIDOS	42
FIGURA 6. CLOSURES DE GROOVY	45
FIGURA 7. MÉTODOS EN GROOVY CON VARIOS PARÁMETROS	45
FIGURA 8. USO DE HTML5 ETIQUETA VIDEO	46
FIGURA 9. USO DE JQUERY	46
FIGURA 10. USO DE JQUERY-UI	47
FIGURA 11. USO DE JQUERY-DATATABLES	48
FIGURA 12. ARQUITECTURA GRAILS	49
FIGURA 13. SPRINGSOURCE TOOL SUITE	51
FIGURA 14. FIREBUG CON CÓDIGO HTML Y CSS	52
FIGURA 15. FIREBUG COMO DEPURADOR DE JAVASCRIPT	52
FIGURA 16. VIDEO DOWNLOADHELPER	53
FIGURA 17. USO DE GIMP2	53
FIGURA 18. NOMENCLATURA DE DIAGRAMA DE USO	73
FIGURA 19. PATRÓN DE DESCRIPCIÓN DE CASO DE USO	74
FIGURA 20. DIAGRAMA DE CASO DE USO DEL ADMINISTRADOR DEL SISTEMA	75
FIGURA 21. ARQUITECTURA CLIENTE SERVIDOR	83
FIGURA 22. MODELO VISTA CONTROLADOR	84
FIGURA 23. EJEMPLO DE UML	86
FIGURA 24. UML - MODELO RELACIONAL	87
FIGURA 25. UML ASIGNATURAS	88
FIGURA 26. UML SUBASIGNATURAS	88
FIGURA 27. UML VIDEOS	88
FIGURA 28. UML PREGUNTAS	89
FIGURA 29. LEYENDA IMAGEN	89
FIGURA 30. LEYENDA TEXTO	89
FIGURA 31. LEYENDA BOTONES	90
FIGURA 32. LEYENDA INTRODUCIR TEXTO	90
FIGURA 33. LEYENDA VIDEO	90
FIGURA 34. LEYENDA ENLACE	90
FIGURA 35. LEYENDA CAMPO EDITABLE	90
FIGURA 36. LEYENDA DESPLEGABLE	90
FIGURA 37. LEYENDA SELECCIÓN RADIO BUTTONS	90
FIGURA 38. PÁGINA PRINCIPAL APLICACIÓN	91
FIGURA 39. LISTADO DE BÚSQUEDA GENERAL	91
FIGURA 40. LISTADO DE BÚSQUEDA POR SUBASIGNATURAS	92
FIGURA 41. REPRODUCCIÓN DEL VIDEO	93
FIGURA 42. VENTANA DE PREGUNTAS	93
FIGURA 43. COMPROBACIÓN DE RESPUESTA	94
FIGURA 44. VOLVER A LA REPRODUCCIÓN DEL VIDEO	95
FIGURA 45. PÁGINA PARA EL USUARIO COLABORADOR	95
FIGURA 46. FORMULARIO DE COLABORACIÓN	96
FIGURA 47. MENSAJE DE ENVÍO DE EMAIL	96
FIGURA 48. PANTALLA PRINCIPAL DE ADMINISTRADOR	97

FIGURA 49. PANTALLA GESTIONAR.....	97
FIGURA 50. LISTADO DE ASIGNATURAS.....	98
FIGURA 51. ALTA DE ASIGNATURAS	98
FIGURA 52. CONSULTA DE ASIGNATURAS.....	99
FIGURA 53. EDITAR ASIGNATURAS.....	99
FIGURA 54. ELIMINAR ASIGNATURAS	100
FIGURA 55. LISTADO DE SUBASIGNATURAS	101
FIGURA 56. ALTA DE SUBASIGNATURAS.....	101
FIGURA 57. CONSULTA DE SUBASIGNATURAS	102
FIGURA 58. EDITAR SUBASIGNATURAS.....	102
FIGURA 59. ELIMINAR SUBASIGNATURAS.....	103
FIGURA 60. LISTADO DE VIDEOS.....	103
FIGURA 61. ALTA DE VIDEOS	104
FIGURA 62. CONSULTA DE VIDEOS.....	104
FIGURA 63. EDITAR VIDEOS	105
FIGURA 64. ELIMINAR VIDEOS	105
FIGURA 65. LISTADO DE PREGUNTAS	106
FIGURA 66. ALTA DE PREGUNTAS.....	106
FIGURA 67. CONSULTA DE PREGUNTAS	107
FIGURA 68. EDITAR PREGUNTAS	107
FIGURA 69. ELIMINAR PREGUNTAS.....	108
FIGURA 70. ESTRUCTURA DE LA APLICACIÓN	112
FIGURA 71. CLASE DE DOMINIO VIDEOS	113
FIGURA 72. CONTROLADOR VIDEOS - PARTE1	115
FIGURA 73. CONTROLADOR VIDEOS - PARTE2	116
FIGURA 74. CONTROLADOR VIDEOS - PARTE3	117
FIGURA 75. CONTROLADOR VIDEOS - PARTE4	118
FIGURA 76. CONTROLADOR VIDEOS - PARTE5	119
FIGURA 77. CONTROLADOR VIDEOS - PARTE6	120
FIGURA 78. VISTA GSP DE VIDEOS - PARTE1.....	125
FIGURA 79. VISTA GSP DE VIDEOS - PARTE2.....	126
FIGURA 80. EJEMPLO DE ETIQUETA G: EACH USADAS EN LAS VISTAS GSP	127
FIGURA 81. EJEMPLO DE ETIQUETA G: RENDER USADAS EN LAS VISTAS GSP	128
FIGURA 82. SERVICIO HOME SERVICE	129
FIGURA 83. USO DE JAVASCRIPT PARA VIDEOS	130
FIGURA 84. PRUEBA UNITARIA SOBRE BÚSQUEDA GENERAL.....	133
FIGURA 85. RESULTADO DE PRUEBA SOBRE BÚSQUEDA GENERAL O POR SUBASIGNATURAS	134
FIGURA 86. PRUEBA UNITARIA SOBRE FORMULARIO DE COLABORACIÓN.....	134
FIGURA 87. PRUEBA UNITARIA EN REPRODUCCIÓN DE VIDEOS.....	135
FIGURA 88. PRUEBA UNITARIA EN PÁGINA DE ADMINISTRADOR.....	136
FIGURA 89. RESULTADO DE PRUEBA UNITARIA EN PÁGINA DEL ADMINISTRADOR.....	137
FIGURA 90. PRUEBA UNITARIA EN GESTIONAR ASIGNATURAS	137

Índice de Tablas

TABLA 1. PLANIFICACIÓN Y ESTIMACIÓN DEL PROYECTO.....	17
TABLA 2. PRESUPUESTO ECONÓMICO - COSTE PERSONAL	20
TABLA 3. SUELDO DEL PERFIL INGENIERO	20
TABLA 4. PRESUPUESTO ECONÓMICO.....	21
TABLA 5. EVOLUCIÓN DE HTML	37
TABLA 6. PATRÓN DE REQUISITO	56
TABLA 7. REQUISITO FUNCIONAL - ALTA DE ASIGNATURAS	57
TABLA 8. REQUISITO FUNCIONAL - BAJA DE ASIGNATURAS	58
TABLA 9. REQUISITO FUNCIONAL - MODIFICACIÓN DE ASIGNATURAS	58
TABLA 10. REQUISITO FUNCIONAL - LISTADO DE ASIGNATURAS.....	59
TABLA 11. REQUISITO FUNCIONAL - CONSULTA DE ASIGNATURAS	59
TABLA 12. REQUISITO FUNCIONAL - ALTA DE SUBASIGNATURAS	60
TABLA 13. REQUISITO FUNCIONAL - BAJA DE SUBASIGNATURAS.....	60
TABLA 14. REQUISITO FUNCIONAL - MODIFICACIÓN DE SUBASIGNATURAS.....	61
TABLA 15. REQUISITO FUNCIONAL - LISTADO DE SUBASIGNATURAS	61
TABLA 16. REQUISITO FUNCIONAL - CONSULTA DE SUBASIGNATURAS.....	62
TABLA 17. REQUISITO FUNCIONAL - ALTA DE VIDEOS.....	62
TABLA 18. REQUISITO FUNCIONAL - BAJA DE VIDEOS	63
TABLA 19. REQUISITO FUNCIONAL - MODIFICACIÓN DE VIDEOS	63
TABLA 20. REQUISITO FUNCIONAL - LISTADO DE VIDEOS.....	64
TABLA 21. REQUISITO FUNCIONAL - CONSULTA DE VIDEOS	64
TABLA 22. REQUISITO FUNCIONAL - ALTA DE PREGUNTAS.....	65
TABLA 23. REQUISITO FUNCIONAL - BAJA DE PREGUNTAS.....	65
TABLA 24. REQUISITO FUNCIONAL - MODIFICACIÓN DE PREGUNTAS	66
TABLA 25. REQUISITO FUNCIONAL - LISTADO DE PREGUNTAS	66
TABLA 26. REQUISITO FUNCIONAL - CONSULTA DE PREGUNTAS	67
TABLA 27. REQUISITO FUNCIONAL - BÚSQUEDA GENERAL.....	67
TABLA 28. REQUISITO FUNCIONAL - BÚSQUEDA DE SUBASIGNATURAS	68
TABLA 29. REQUISITO FUNCIONAL - INTERACCIÓN CON VIDEOS.....	68
TABLA 30. REQUISITO FUNCIONAL - FORMULARIO DE COLABORACIÓN	69
TABLA 31. REQUISITO NO FUNCIONAL - FUNCIONAMIENTO CON INTERNET	69
TABLA 32. REQUISITO NO FUNCIONAL - FUNCIONAMIENTO CON NAVEGADOR WEB	70
TABLA 33. REQUISITO NO FUNCIONAL - COMPATIBILIDAD CON HTML5.....	70
TABLA 34. REQUISITO NO FUNCIONAL - COMPATIBILIDAD CON MOZILLA FIREFOX	70
TABLA 35. REQUISITO NO FUNCIONAL - COMPATIBILIDAD CON GOOGLE CHROME.....	70
TABLA 36. REQUISITO NO FUNCIONAL - COMPATIBILIDAD CON INTERNET EXPLORER.....	71
TABLA 37. REQUISITO NO FUNCIONAL - COMPATIBILIDAD CON SAFARI.....	71
TABLA 38. REQUISITO NO FUNCIONAL - FRAMEWORK.....	71
TABLA 39. REQUISITO NO FUNCIONAL - LENGUAJE DE PROGRAMACIÓN	71
TABLA 40. REQUISITO NO FUNCIONAL - ENTORNO DE DESARROLLO	72
TABLA 41. REQUISITO NO FUNCIONAL - BASE DE DATOS	72
TABLA 42. REQUISITO NO FUNCIONAL - SERVIDOR	72
TABLA 43. REQUISITO NO FUNCIONAL - FORMATO DE VIDEOS PARA HTML5.....	72
TABLA 44. DESCRIPCIÓN DEL CASO DE USO - GESTIÓN DE ASIGNATURAS	75
TABLA 45. DESCRIPCIÓN DEL CASO DE USO - GESTIÓN DE SUBASIGNATURAS.....	76
TABLA 46. DESCRIPCIÓN DEL CASO DE USO - GESTIÓN DE VIDEOS	76

TABLA 47. DESCRIPCIÓN DEL CASO DE USO - GESTIÓN DE PREGUNTAS	77
TABLA 48. DIAGRAMA DE CASO DE USO DEL USUARIO CONSUMIDOR DEL SISTEMA	77
TABLA 49. DESCRIPCIÓN DEL CASO DE USO - CONSULTA GENERAL	78
TABLA 50. DESCRIPCIÓN DEL CASO DE USO - CONSULTA POR SUBASIGNATURAS	79
TABLA 51. DESCRIPCIÓN DEL CASO DE USO - INTERACTÚA CON VIDEOS	80
TABLA 52. DIAGRAMA DE CASO DE USO DEL USUARIO COLABORADOR CON EL SISTEMA	80
TABLA 53. DESCRIPCIÓN DEL CASO DE USO - RELLENAR FORMULARIO DE COLABORACIÓN	81
TABLA 54. FRAMEWORK CON ARQUITECTURA MVC	84

CAPÍTULO 1

1. Introducción y Objetivos

En este capítulo del documento se detallan de forma breve los objetivos y motivaciones que han incentivado a realizar el presente trabajo, la estructura que compone la memoria, además de exponer la planificación temporal, un listado de los recursos utilizados y el presupuesto del proyecto.

1.1 Introducción

Cada vez es más frecuente acudir hoy en día al uso de videotutoriales como herramienta didáctica para el aprendizaje de casi cualquier tema, por este motivo los videos son la finalidad del proyecto, pensando en que el usuario no se limite a la mera reproducción del mismo, si no que se dé la posibilidad de interactuar con ellos en el mismo momento de la reproducción, con el único fin de ofrecer una autoevaluación de los conocimientos adquiridos.

Precisamente por la amplia utilidad que se dan a los videos nos hemos centrado en ellos, aprendiendo a controlar su reproducción para que los usuarios puedan interactuar con él como si de una clase presencial se tratará, el usuario ya no solo ve la reproducción del video hasta llegar al final de su duración, si no que a medida que avanza la secuencia del video puede saber de antemano si está entendiendo el tema explicado a través de una serie de preguntas que aparecen automáticamente a lo largo de la reproducción.

Cuántas veces hemos visto un video que contiene la explicación de lo que estamos buscando y muchas veces hemos tenido que hacer pausas constantes para tratar de asimilar lo explicado e incluso hacernos preguntas a nosotros mismo sin saber si las estamos respondiendo de forma correcta o incorrecta, atendiendo a estas necesidades se ha desarrollado esta aplicación que permite al usuario interactuar con el video y sea el propio sistema quien se encargue de hacer las preguntas oportuna en los tiempos precisos proporcionando de esta manera al usuario una idea de si el video es útil o no para su aprendizaje.

Podemos decir en cierta forma que el usuario tiene el control de la reproducción normal del video puesto que esto está condicionada por el tipo de contestaciones que el usuario de a las preguntas que se muestren.

El sistema va dirigido principalmente a estudiantes que se ven en la necesidad de consultar videos para comprender mejor o reforzar algún tema tratado en sus horas de clases, aún más podemos pensar en que el sistema puede ser usado por cualquier persona que desee aprender.

1.1.1 Contexto y Motivación

El uso de Videos para el desarrollo de la aplicación emerge debido a la enorme utilidad que tienen hoy en día en varios tipos de sectores y no solo como método de aprendizaje. Detallamos a continuación algunas razones por la que los videos son utilizados en una amplia gama de actividades.

- Lo visual prima sobre los textos, ya que puede comprender mejor la explicación de un tema sin necesidad de mover el mouse.
- Las explicaciones por videos atrae el interés de los internautas.
- Ayudan a retener la información de una manera más fácil.
- El video es entretenido ya que está relacionado con el mundo del entretenimiento y el ocio.
- Pueden ser una opción complementaria para las horas en clase y para las academias de aprendizaje.

1.1.2 Objetivos

El objetivo principal de este Proyecto Fin de Carrera, es desarrollar una aplicación Web que sirva como método interactivo de aprendizaje para el usuario que desean instruirse por su propia cuenta haciendo uso de elementos multimedia, como ya mencionamos anteriormente nos centramos en el elemento multimedia más utilizado, los Videos. Para trabajar con ellos es imprescindible conocer la forma de controlar la reproducción, pausa, avance o retroceso de un video y de esta manera realizar las operaciones necesarias para mostrar las preguntas a ser contestadas por el usuario siendo las respuestas un condicionante importante para llegar al final de la reproducción.

La creación de este sistema de ayuda interactiva para el aprendizaje de asignaturas puede servir también como método de autoevaluación para el usuario, es decir permite saber al usuario si el video le es de utilidad para reforzar su conocimiento adquiridos o por el contrario el video no cumple con las expectativas esperadas por él.

Otro de los objetivos que tiene el sistema es la creación de un tipo de comunidad colaboradora bidireccional.

- Por una lado: nosotros colaborar con las personas que necesitan hacerse conocer a través de los videos publicados en la web, bien sea para poder impartir clases personales o simplemente con el mero hecho de que esta actividad les sirva de ayuda para su futura profesión como docente, en el caso de no serlo y en el caso de serlo les ayuden a mejor como profesores.
- Y por otro lado: el sistema necesita de las personas dispuestas a cedernos sus videos para disponer de una amplia biblioteca, este punto es muy importante puesto que sin los videos la aplicación no tendría ningún sentido.

Se pretende mediante la **Interacción** de los videos el mismo momento de la reproducción que la aplicación ofrezca un elemento diferenciador al resto de Webs que emplean videos para el aprendizaje.

Finalmente los objetivos futuros que abren camino a nuevas funcionalidades o desarrollos basados en el proyecto implementado se detallan el Capítulo 8 del presente documento.

1.1.3 Estructura de la Memoria

Este documento sigue la siguiente estructura de contenidos:

Introducción y objetivos

En este capítulo se incluye el contexto y motivación del proyecto, los objetivos marcados y la estructura general de la memoria. La gestión del proyecto se detallan en este capítulo mediante la planificación temporal de las diferentes tareas que componen el proyecto, un listado de los recursos hardware y software. Finalizando con el presupuesto empleado.

Estado del Arte

Capítulo en el que se describe la importancia de las TIC en la educación y los elementos multimedia como herramientas de aprendizaje, se justifica además la elección de los video como elemento didáctico para aprender, continuando con la descripción de los módulos que conforman el sistema, también se detalla la evolución que ha tenido HTML a lo largo de los años como herramienta útil hoy en día para el manejo de videos. Para terminar este capítulo se describen algunas Webs que hacen uso de videos como métodos de aprendizaje y las ventajas que ofrecen el uso de los mismos.

Tecnologías y herramientas

Este capítulo presenta una breve descripción de la situación actual de las diferentes tecnologías usadas en los desarrollos web, así como los diferentes lenguajes que se pueden usar, y el que finalmente emplearemos para elaborar el proyecto. Se explica además las distintas herramientas que necesitaremos durante todo el desarrollo de nuestro trabajo.

Análisis del sistema

Capítulo en el que se identifica los escenarios y usuarios a los que están destinados la aplicación, así como la especificación de los requisitos tanto funcionales como no funcionales, y para terminar se especifica los casos de usos necesarios.

Diseño de la aplicación

En esta etapa se especifica la arquitectura que va a seguir la aplicación desarrollada, el modelo de datos utilizado, las diferentes interfaces de usuario, los controladores y servicios de la aplicación.

Implementación de la aplicación

Capítulo en el que se explica las secciones de código que, de forma ilustrativa, aporten una visión completa de cómo se ha desarrollado la aplicación.

Pruebas

Se detallan algunas de las diferentes pruebas utilizadas para verificar que la funcionalidad desarrollada es correcta.

Conclusiones y Trabajos Futuros

Se analizaran los resultados obtenidos del desarrollo del proyecto, los objetivos cumplidos, los conocimientos adquiridos y los aspectos a tener en cuenta en el futuro con el objetivo de evolucionar o mejorar ciertos aspectos de la aplicación.

Biografía y Referencias

Se incluyen las fuentes de información, incluyendo los recursos electrónicos o artículos consultados durante la elaboración del proyecto.

1.1.4 Planificación Temporal

Para la planificación del presente proyecto se han creado bloques de trabajo que agrupan cada una de las tareas realizadas, a continuación se representa mediante una tabla los bloques de los que consta el proyecto, así como las tareas llevadas a cabo en cada uno de ellos.

BLOQUES	TAREAS	FECHA INICIO	FECHA FIN	ESTIMACIÓN
Evaluación y Análisis	Evaluación del Sistema que se quiere desarrollar	21/08/2015	27/08/2015	8h
	Análisis de sistema ha desarrollar	28/08/2015	3/09/2015	8h
	Obtención de los requisitos	4/09/2015	09/09/2015	8h
	Detalles de los casos de uso	10/09/2015	11/09/2015	8h
Gestión	Definir la gestión del proceso de desarrollo	14/09/2015	16/09/2015	8h
	Decidir los pasos para la gestión de las tareas del proyecto	17/09/2015	23/09/2015	8h
	Redactar la planificación	24/09/2015	25/09/2015	8h
	Especificar el presupuesto	28/09/2015	28/09/2015	8h
Diseño	Especificar el diseño de la aplicación	29/09/2015	01/10/2015	8h
	Definir la arquitectura de la aplicación	02/10/2015	06/10/2015	8h
	Modelar el diseño del modelo de datos	7/10/2015	9/10/2015	8h
	Diseño de la Interfaz	12/10/2015	15/10/2015	8h
Pruebas	Pruebas unitarias realizadas	16/10/2015	19/10/2015	8h
	Resultado de las Pruebas y Evaluación	20/10/2015	20/10/2015	8h
Conclusiones	Evaluación y difusión de Resultados	21/10/2015	21/10/2015	8h
Documentación	Elaboración de la memoria	22/10/2015	03/11/2015	8h

Tabla 1. Planificación y Estimación del Proyecto

Diagrama de Gantt

Se ilustra de manera gráfica la planificación de las diferentes tareas del proyecto, además de observar cómo se distribuyen a lo largo del tiempo.

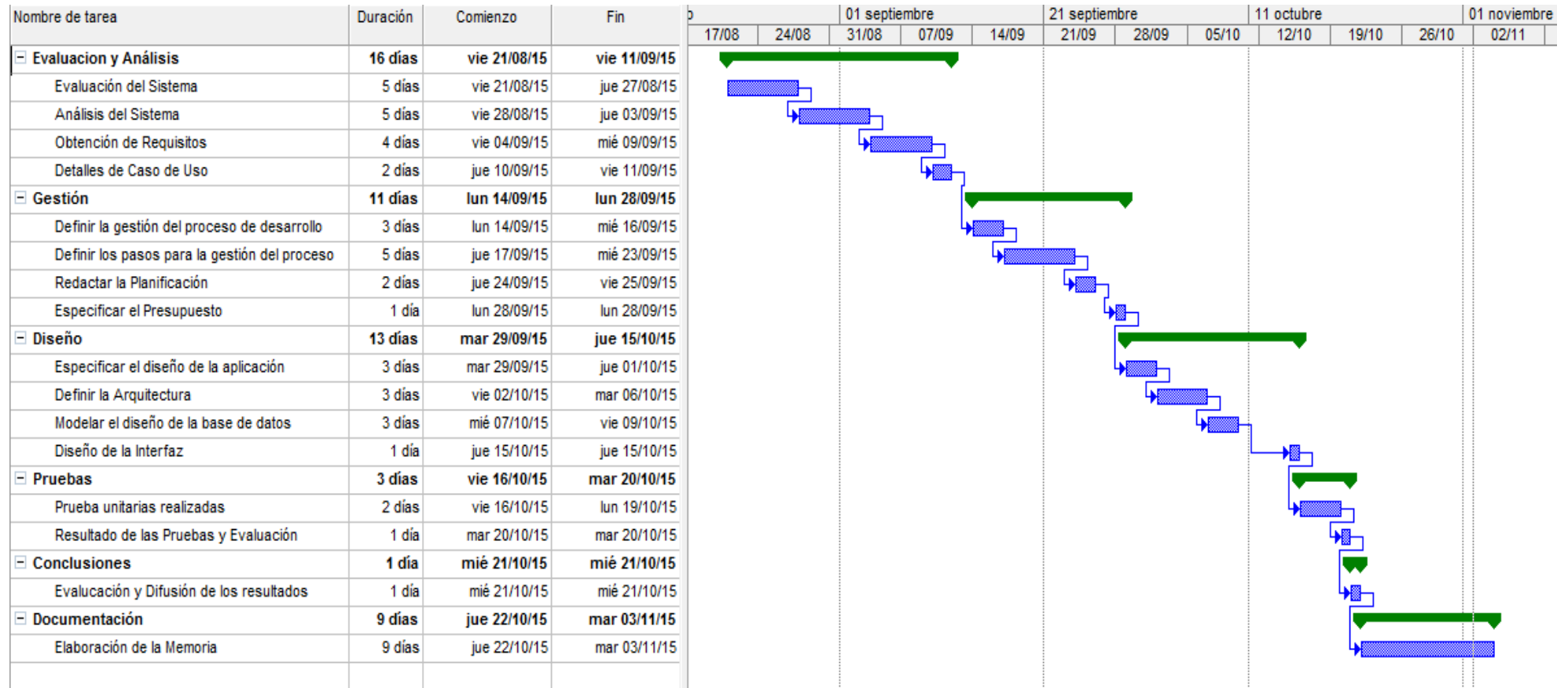


Figura 1. Diagrama de Gantt

1.1.5 Recursos

Para llevar a cabo el proyecto ha sido necesario los siguientes recursos:

Recursos Hardware

- Ordenador Portátil

Recursos Software

- Entorno de desarrollo STS
- JDK (Java development kit): Kit de desarrollo de Java
- MySQL Workbench herramienta que permite la creación de las bases de datos a utilizar
- Youtube
- GIMP
- Video DownloadHelper: esta herramienta es de gran utilidad para descargarnos a nuestro equipo los videos con lo que trabajaremos en la aplicación.
- JQuery
- JQueryTable
- JQuery-UI
- Simple Captcha: Plugins de Grails
- Mail: Plugins de Grails
- Firebug
- Navegadores Firefox, Internet Explorer y Google Chrome
- Microsoft Office 2007

El coste de estos recursos no han sido tomados en cuenta para realizar el presupuesto del proyecto ya que tanto el recurso software, ha sido obtenido a través de internet de manera gratuita, como el recurso hardware lo hemos tenido disponible desde un principio, no ha sido necesario comprarlo para uso exclusivo del proyecto.

1.1.6 Presupuesto

Se detalla el presupuesto para el desarrollo de la aplicación detallando los costes que implica el proceso.

Tarea	Duración (jornadas)	Esfuerzo (h/jornada)	Total en Horas
Evaluación y Análisis			
Perfil Ingeniero	16	8	128
Gestión			
Perfil Ingeniero	11	8	88
Diseño			
Perfil Ingeniero	13	8	104
Pruebas			

Perfil Ingeniero	3	8	24
Conclusiones			
Perfil Ingeniero	1	8	8
Documentación			
Perfil Ingeniero	9	8	72
Total Horas			424 hrs

Tabla 2. Presupuesto Económico - Coste Personal

Para calcular el coste de la persona se toma como referencia el sueldo mensual bruto del perfil involucrado, acorde con el mercado actual.

Perfil	Sueldo bruto €/año	Sueldo bruto €/mes
Perfil Ingeniero	24.000 €/año	2.000 €/mes

Tabla 3. Sueldo del Perfil Ingeniero

Se muestra a continuación un resumen del presupuesto obtenido.

PRESUPUESTO DEL PROYECTO	
1.- Autor	
Paulina Alexandra Silva Simbaña	
2.- Departamento	
Informática	
3.- Descripción del Proyecto	
- Título : <u>Aplicación Web para el Aprendizaje Mediante el Uso Videos</u>	
- Duración (meses): 1.76	
4.- Presupuesto Total	
3532 Euros (Tres mil quinientos treinta y dos euros)	
5.- Desglose Presupuestario	
424 horas de desarrollo corresponden a 1.766 mes	

$$\frac{total\ horas}{8h/día} = número\ de\ días;$$

$$\frac{424\ horas}{8\ horas/día} = 53\ días$$

$$\frac{número\ de\ días}{30\ día/mes} = número\ de\ meses;$$

$$\frac{53\ días}{30\ días\ /mes} = 1.766\ mes$$

Coste Personal : 1.766mes x 2000€/mes = 3532€

Tabla 4. Presupuesto Económico

CAPÍTULO 2

2. Estado del Arte

Este capítulo tiene como principal finalidad describir la posibilidad que ofrecen los videos como medio de enseñanza de gran utilidad, basándonos en la importancia de las TIC en el mundo de la Educación.

2.1 Importancia de la TIC en la educación

Gracias al importante avance que ha tenido la tecnología en los últimos años, no solo en la Infraestructura de Redes, sino también a los avances tecnológicos aplicados a los dispositivos portátiles, a los que se incorpora una gran cantidad y variedad de funcionalidades, y a los que son accesibles fácilmente por casi todo el mundo de la sociedad actual, el grupo conocido como Tecnologías de la Información y Comunicación (TIC) apunta al estudio de todo lo que es el manejo de datos, envío y recepción de paquetes de información.

En el mundo moderno en el que vivimos estamos tanto mayores y niños acostumbrados a utilizar dispositivos portátiles como es el caso de la telefonía móvil, aparato que antiguamente era usado tan solo para el envío y recepción de llamadas, ahora tiene una amplia variedad de funciones multimedia además de utiliza redes móviles de alta velocidad.

Precisamente por la utilidad que ha tomado en la vida cotidiana se ha prestado mayor interés como herramienta en el Ámbito Educativo, ya que es posible acceder a una gran cantidad de material didáctico, convirtiéndose en una instrumento de búsqueda para aplicaciones educativas, claro está que hay controversias con las personas que prefieren lo tradicional de los libros y la escritura manuscrita.

Debido al avance vertiginoso de la tecnología es necesario distribuir versiones digitales de muchos textos escolares para que puedan ser utilizados mediante un dispositivo apto para su correcta visualización, en este caso nos puede servir un lector de libros digitales. Al digitalizar un libro de texto surge la necesidad de asignar a cada alumno un dispositivo portátil como puede ser un netbooks escolar y de esta manera hacer que se familiarice con los ordenadores para el uso de contenidos digitales.

Se describen a continuación seis razones de peso que se desprenden de la utilización de las TIC por parte de los alumnos.

1. **Competencia digital e informacional del alumnado.** Se ha convertido en una necesidad básica que los alumnos y el resto de ciudadanos hagan un buen uso de la tecnología digital, permitiendo así que desarrollen sus competencias digitales e informacionales, que les puede dar la posibilidad de aprovechar y sacar lo máximo que ofrecen las TIC para desarrollarse mejor, adaptarse mejor y encontrar mejores trabajos.
2. **Productividad.** Una consecuencia beneficiosa del buen uso de las TIC es hacernos más productivos. Esta productividad se vería reflejada en el hecho de realizar tareas que antes era impensables, hacer el mismo trabajo en menos tiempo, e incluso mejorar una trabajo ya realizado. La posibilidad que ofrece Internet en realizar actividades a distancia, no hace falta salir de casa, hace que esto permita hacer dichas actividades lo más rápido posible pensando además en los ahorros económicos que supondría tanto el desplazamiento como la lentitud en dar una respuesta.
Los profesores y alumnos ven una ventaja beneficiosa en la Educación ya que el hecho de hacer un buen uso de las TIC aumenta la productividad, les ayuda a hacer muchos trabajos: escribir, calcular, crear bases de datos, buscar información, estudiar,

comunicarse, compartir, publicar trabajos con otros, etc. Todos estos trabajos son realizados siempre haciendo uso de nuestra inteligencia que sin duda es irremplazable, el dispositivo que utilicemos para realizar las actividades antes descritas solamente hace lo que nosotros le decimos, el no piensa por sí mismo.

3. **Recurso didáctico para innovar.** Con el fin de reducir el fracaso escolar las TIC (hardware, software y material multimedia) se han convertido en un enorme apoyo para realizar cosas nuevas, aplicando nuevas tecnologías con el principal objetivo de ofrecer un método innovar a los alumnos e incentivarles en las ganas de aprender.
4. **Actualización curricular.** La influencia que ejerce las TIC en casi todas las actividades y en los distintos campos es conocido por casi todos, puesto que han modificado la forma de trabajar introduciendo herramientas tanto generales como específicas en cada campo de conocimiento. Por lo tanto las TIC nos permiten evolucionar en la manera de trabajar, actualmente nada se hace como lo hacíamos antes.
5. **Aprendizaje continuo.** Para las personas que hacen uso de teléfono móvil, tablets digitales, o de un ordenador, acudimos a ellos siempre para consultar temas de los que tenemos desconocimiento, esto hace que la información de todo el mundo este a nuestra entera disposición. El hecho de tener estos dispositivos a nuestro alcance hace que necesitemos aprender constantemente algo nuevo, o simplemente informarnos de algo que ignoramos. Las respuestas que obtengamos a nuestras preguntas deben ser elegidas de tal manera que sepamos discernir lo verdadero de lo falso, esto lo podemos aprender en la escuela o en trabajo, o dedicando muchas horas a la práctica, ya que como dice el dicho " La práctica hace al maestro". Debemos procurar hacer que las consultas realizadas a estos dispositivos nos ayuden a aprender y entender y no permitir que solo sean conocimientos memorizados en su momento.
6. **Comunicación y multipresencialidad virtual.** La comunicación entre personas que dispongan de una identidad digital está garantizada por las TIC, además de permitir hacer cosas de manera simultánea en lugares lejanos entre sí. Con las TIC podemos enseñar, aprender, ayudar, pedir ayuda, compartir el conocimiento haciendo que esto nos haga sentirnos bien con nosotros mismos.

Trabajar con las TIC permite a los alumnos que desarrollen:

- **Autonomía:** Las TIC nos ponen casi todo a nuestro alcance, nos hacen más autónomos.
- **Toma de decisiones:** Aplicar criterios para saber lo que debemos elegir en Internet.
- **Desarrollo de criterio:** Aplicaremos criterios para tomar decisiones, que irán bien o no, y así iremos aprendiendo.
- **Aprendizaje a partir del error:** Debido a las innumerables alternativas que ofrece internet es fácil cometer errores a la hora de elegir alguna de ellas, habrán momentos que elijamos bien y otro no, es necesario aprender de los errores.
- **Desarrollo de habilidades sociales:** Estamos mucho más en contacto con otros a través de e-mail, redes sociales, etc.

- **Trabajo colaborativo:** Las TIC nos ofrecen entornos "ad hoc" para el trabajo colaborativo a distancia.
- **Compartir y participar:** Las TIC ha hecho posible crear muchos entornos en los que podamos compartir información, material digital, etc. El hecho de compartir digitalmente debe dar lugar a que esta actividad se propague siempre respetando al propietario de la información.
- **Creatividad:** Tener mucha información nos ayuda a ser creativos (la información es materia prima para la creatividad). Tener muchas herramientas diversas (como nos ofrecen las TIC) nos permite encontrar nuestro "talento" y poder ser creativos desarrollando "nuestro talento".
- **Perseverancia:** Las TIC e Internet nos ofrecen tanta información, herramientas y oportunidades que si persistimos conseguiremos y encontraremos la alternativa que mejor se adapte a nuestras necesidades.
- **Lenguas:** Al disponer en internet información de todos los lugares del mundo, nos encontramos con la barrera del idioma. Cuántas veces hemos hallado páginas, que se adecuan a nuestra búsqueda, escritas en otra lengua, he aquí otro motivo para aprender Idiomas ya que si lo hacemos disponemos de más información eliminando la barrera del idioma y no limitándonos a las páginas escritas solamente en nuestro idioma natal. ¿Quién o Qué puede ayudarnos en el aprendizaje de otras lenguas?, sin duda Internet.

2.2 Multimedia como herramienta educativa

Se puede definir al término "multimedia" como la unión de dos o más medios para transmitir información tales como texto, imágenes, animaciones, sonido y video que llega al usuario a través de un medio electrónico.

En internet existe una amplio abanico de herramientas que estimulan el aprendizaje, como pueden ser videos, gráficas, actividades interactivas, libros digitales, etc, cuyo objetivo es buscar optimizar la presentación de los contenidos y la forma en que los usuarios interactúan, estimulando de esta manera los sentidos como el tacto, el oído y especialmente el cerebro quién es el responsable de aprender.

El estallido del multimedia ha modificado la forma de ofrecer los contenidos y el modo de motivar a los alumnos en el proceso de aprendizaje, centrándose de esta manera en proponer avances tecnológicos en el mundo multimedia de la Educación.

Atendiendo las necesidades de los profesores en cuanto a la creación de materiales educativos multimedia que le faciliten la labor de la enseñanza, se permite en este sentido al profesor estar involucrado en el diseño de estos materiales cuyo resultado sean reforzar la enseñanza mediante el uso de recursos gráficos y audiovisuales para consolidar los contenidos y además de esta manera se consiga motivar a la participación activa de los alumnos.

El personal docente puede hacer uso de repositorios que existen en internet con herramientas multimedia, como apoyo académico, estos repositorios permiten libre utilización al mismo tiempo que son avalados por un colectivo cualificado cumpliendo las características de calidad que se requieren para ser publicados.

Tal y como se ha indicado una herramienta multimedia sirve para que el docente lo utilice como apoyo académico, para que aumente la atracción hacia un aprendizaje debido a que el alumno está en contacto directo con el material y las actividades siempre dirigidas por el docente con fines formativos.

2.2.1 Herramientas para crear actividades educativas multimedia

- El docente de la actualidad debe pensar continuamente en la mejor manera de innovar sus clases, para que sean más atractivas, motivadoras e interactivas. Por ello es que surgen diversos software que permiten crear recursos y actividades educativas multimedia que permite mejorar significativamente el trabajo del profesorado en las aulas.

Se presenta una lista de 10 software libre que permitirá crear recursos y actividades educativas multimedia.

Nombre	Descripción
Ardora	Ardora es una herramienta sencilla que permite crear tanto actividades en formato HTML como páginas multimedia: crucigramas, sopas de letras, galerías de imágenes, reproductores, etc. Cuenta con una versión portable y está disponible en varias lenguas. En su web se muestran ejemplos , los ficheros de descarga y varias ayudas.
Constructor	Constructor crea contenidos educativos digitales y funciona en Windows y Linux, en local o en un servidor. Tiene 53 modelos de actividades, desde juegos hasta aplicaciones y permite la inclusión de applets. Genera ODEs (Objetos Digitales Educativos) en paquetes Zip. En la web de la Consejería de Educación y Cultura de la Junta de Extremadura hay un paquete de recursos para ver y descargar que se pueden usar en las aulas y también en los propios domicilios para fomentar la autonomía de aprendizaje del alumnado.
Cuadernia	Cuadernia es la herramienta de creación de contenidos digitales educativos de la Consejería de Educación y Ciencia de Castilla-La Mancha. Con esta herramienta se pueden crear recursos reutilizables, cuenta con un editor de cuadernos digitales, y una biblioteca de ODEs para Infantil, Primaria y Secundaria. Los recursos son visualmente atractivos y motivadores para el alumnado; están catalogados siguiendo el estándar LOM-ES 1.0 y añade además la herramienta Cuaderna Catalogación para catalogar un Objeto Digital Educativo (ODE).
Edilim	EdiLim es el editor de libros de LIM , un agradable entorno para la creación de materiales educativos digitales que se complementa con un visualizador y un fichero xml, el libro. Desde la web del proyecto se puede acceder a la biblioteca de libros, donde encontrarás varios materiales para el área de inglés .
ExeLearning	Exe learning es el editor XHTML para la creación de contenidos para elearning. Fácil de utilizar y bastante flexible para exportar, importar y

	reutilizar contenidos, permite crear curso digitales completos. Se puede empaquetar los contenidos como SCORM para después llevarlos a la plataforma con la que trabajes, por ejemplo, Moodle.
Hotpotatoes	Hot Potatoes es quizá la patata caliente más popular en el campo de creación de recursos educativos digitales. Cuenta con seis aplicaciones que permiten realizar actividades de opción múltiple, de respuestas cortas, crucigramas, ejercicios de rellenar huecos, ordenar y asociar. Funciona en Windows y Linux y necesitas tener instalado Java en tu equipo
JClic	JClic sirve para realizar diferentes actividades educativas digitales: rompecabezas, ejercicios de asociación, entre otros. Está desarrollado sobre plataforma Java y sus actividades se visualizan a través de un applet. También se pueden descargar en local y guardar en la biblioteca de actividades. En su web, cuenta con explicaciones sobre las dos formas de acceder a las actividades, además de páginas de ejemplos a los que se puede acceder mediante un buscador y que están enfocadas a Infantil, Primaria, ESO y Bachillerato.
Lams	Lams es una herramienta de autor para la creación de actividades colaborativas. En un entorno de autor, el profesor puede diseñar secuencias de contenidos digitales educativos dirigidas al gran grupo. Cuenta con una pantalla de gestión que permite conocer el progreso del alumnado en la realización de una secuencia de actividades diseñada con la herramienta, analizar las dificultades que se han encontrado al hacer la tarea propuesta y ver cómo están enfrentándose a la tarea.
Malted	Malted es quizá la herramienta de autor más completa para la creación de contenidos educativos digitales para idiomas. Es la única herramienta de autor que permite la grabación de voz del alumnado, digitalizando así también el trabajo en el aula de las competencias comunicativas, y que además permite su posterior evaluación por parte del profesorado, ya que las grabaciones permanecen en el equipo en el que se han realizado. Para visualizar sus recursos TIC online es necesario tener instalados en el equipo la máquina virtual de Java y el plugins Malted Web 2.0. También se puede trabajar en local ya sea visualizando los materiales mediante cualquier navegador web o mediante el propio navegador independiente de Malted.

2.3 Video - Elemento Multimedia para la Enseñanza

Los avances conseguidos en cuanto a tecnología educativa se refiere invitan a utilizar diversas alternativas con el objeto de favorecer un entorno de enseñanza-aprendizaje adecuado.

El progreso conseguido en los recursos multimedia con el fin de ser utilizados como herramientas didácticas han permitido que los videos, con una adecuada utilización, se hagan más populares por la posibilidad que ofrecen al favorecer la motivación de los alumnos en el aula como a la de contribuir a mejorar el aprendizaje colaborativo.

Al encontrarse los videos en su apogeo, bien sea en el ámbito educativo, o en otro de los distintos campos que se abierto camino, son una opción excelente para alcanzar tácticas formativas innovadoras en todos los ámbitos.

El video es un medio tecnológico que por sus posibilidades expresivas puede ocasionar un alto grado de motivación, convirtiéndose en una herramienta muy apreciada por parte del alumno.

Por si mismo el video se ha convertido en un instrumento valioso en diferentes sectores, podemos darle un valor añadido permitiendo interaccionar sobre el medio, transformándose así en una estrategia de uso más, proporcionando de esta manera tanto al profesor como al alumno una idea sobre el ritmo de enseñanza a seguir para que se adecue a las dificultades de comprensión o retención.

2.3.1. Video como un factor de influencia en la motivación, la satisfacción al aprendizaje exitoso

Se han realizado estudios en el que se determinan como los elementos multimedia han sido una influencia positiva para la motivación, satisfacción y aprendizaje exitoso.

De los números estudios realizados que podemos encontrar en Internet, en el que justifiquen del porque los elementos multimedia se han convertido en un material victorioso para la educación y el aprendizaje hemos escogido la "teoría cognitiva", los siguientes párrafos explicarán de lo que consta esta teoría.

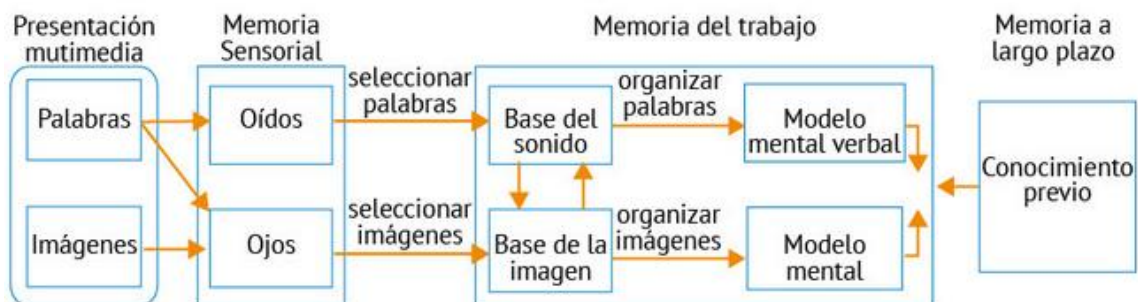


Figura 2. Teoría cognitiva del aprendizaje multimedia (Mayer, 2001)

Los procesos de enseñanza modernos se basan en el uso de materiales didácticos multimedia y de Internet. La teoría cognitiva del aprendizaje multimedia representa una base para la aplicación de la presentación de contenidos educativos multimedia. Esta teoría explica el significado del principio de modalidad en el proceso de aprendizaje, como se muestra en la Figura 2.

Para procesar eficientemente la información multimedia es necesario seleccionar la información relevante y organizarla en un modelo verbal y pictórico. El procesamiento de la información se realiza en dos canales después de recibir la información auditiva y visual que aparece en la memoria de trabajo como modelos de información verbal y pictórica. Un uso racional de los recursos es muy importante, por lo que el diseño de la presentación multimedia es un factor crucial para el resultado del proceso de aprendizaje (Mayer, 2001; Moreno y Mayer, 1999).

La teoría de la carga cognitiva, la investigación educativa y el diseño de enseñanza están estrechamente relacionados, por lo que deben ser analizados en conjunto para lograr los mejores resultados de aprendizaje (De Jong, 2010). La investigación previa también muestra que la eficiencia del proceso de aprendizaje está estrechamente relacionado con la interacción entre las variables motivacionales y cognitivas (Valle, Cabanach, Nunez, Gonzalez-Pienda, Rodríguez, y Piñeiro, 2003).

La tecnología multimedia a menudo ha sido explotada para mejorar la enseñanza y el aprendizaje. Los videos son una herramienta para involucrar la inteligencia verbal (lingüística), visual (espacial), y musical (rítmico) del estudiante en el proceso de aprendizaje, especialmente en el proceso de auto-aprendizaje (Gardner, 2000). Los efectos positivos de las características de los fragmentos de videos (multi-sensoriales, dinámicos y capaces de llamar la atención del espectador) se utilizaron a menudo en educación relacionada con las ingenierías (Marques, 2012).

Se pueden utilizar diferentes métodos de inclusión y diferentes contextos de contenido de video complementario en el proceso de enseñanza. Los profesores están utilizando presentaciones multimedia preparadas para el curso, así como videos adicionales complementarios. Los videos complementarios pueden tener diferentes contextos. Dos de ellos son importantes para este estudio: el primero está estrechamente relacionado con el tema de la clase presentada y el segundo contexto es de entretenimiento, utilizado para captar la atención y el interés de los estudiantes. Los métodos para la inserción y el uso de los diferentes contextos de video complementario en los materiales educativos multimedia son tareas importantes para la investigación con el objetivo de mejorar el proceso de aprendizaje.

En esta investigación, el propósito del uso de material de video complementario es aumentar la atención de los estudiantes sobre el tema de la clase. Con el fin de incrementar la calidad del aprendizaje, se deben alcanzar dos objetivos importantes: reducir la carga cognitiva y aumentar la atención del estudiante. Además de contenido educativo y de entretenimiento, es posible utilizar contenido comercial como video complementario. Aunque el contenido comercial puede contribuir a reducir la carga cognitiva del estudiante, este tipo de contexto potencialmente puede distraer la atención de los estudiantes del tema de la clase y enfocarla en el contenido comercial. Debido a eso, los videos complementarios con contenido comercial han sido menos relevantes para esta investigación que los de contenido educativo y de entretenimiento.

Investigaciones anteriores exploraron los resultados del uso del video como una herramienta educativa (Bravo, Amante-García, Simo, Enache y Fernández, 2011). En este estudio los autores utilizaron videos educativos, con una duración de aproximadamente cuatro minutos, en la plataforma de enseñanza basada en Moodle y canales de YouTube para su reproducción. El estudio mostró que los videos empleados como material de apoyo para el aprendizaje tuvieron un efecto positivo en la percepción de los estudiantes en relación a la mejora de la motivación de aprendizaje. El estudio enfatizó que la definición adecuada del contenido y la cantidad de información proporcionada a través del video complementario es una tarea importante que resolver. La duración de los videos y las metodologías para visualizarlos

influyen en la eficacia del aprendizaje y en la satisfacción de los estudiantes. Un resultado importante de esta investigación es que la transmisión de videos complementarios mejora la capacidad de los estudiantes para aprender de manera autónoma.

Hsin y Cigas (2013) utilizaron videos cortos para mejorar la satisfacción y la motivación de los estudiantes durante un curso introductorio de informática/matemáticas en línea. Alcanzaron un porcentaje significativamente más alto de estudiantes involucrados y sus calificaciones promedio aumentaron. Otro método para involucrar y motivar a los estudiantes es mostrando videos de entretenimiento (Steffes y Duverger, 2012), mostrar videos complementarios con un contexto de entretenimiento al comienzo de la clase se puede utilizar para aumentar el estado de ánimo positivo de los estudiantes. Ambos estudios, Bravo, Amante-García, Simo, Enache y Fernández (2011) y Steffes y Duverger (2012), enfatizan la importancia de un diseño adecuado de los contenidos de video complementario y la metodología para mostrar el video a los estudiantes. Donkor (2011) mostró que el uso de materiales de instrucción basados en video para la enseñanza de habilidades prácticas a distancia también parece prometedor.

Una de las cuestiones clave es la calidad de experiencia lograda con los materiales multimedia presentados. Este problema es más interesante si se combinan los diferentes contextos de los contenidos de video. En el estudio presentado en *Qualinet white paper on definitions of quality of experience*, Le Callet, Moller y Perkis (2012) definieron a la experiencia como la transmisión de la percepción de un individuo y la interpretación que le da ese mismo individuo a uno o varios eventos. Por otro lado, la calidad se define como un resultado del proceso de comparación y juicio de un individuo, que incluye la percepción, la reflexión sobre la percepción, y la descripción de los resultados.

Por último, una definición básica de la calidad de experiencia es:

Calidad de experiencia es el grado de placer o disgusto que presenta el usuario de una aplicación o servicio. Resulta del cumplimiento de sus expectativas con respecto a la utilidad y/o disfrute de la aplicación o servicio en función de la personalidad y el estado actual del usuario. (Le Callet, Moller y Perkis, 2012)

La definición de la calidad de experiencia da un marco para el análisis de la eficiencia de la entrega de contenido de video a través de Internet. Ljubojevic en *The analysis of the users' response to the linear internet video advertising by using QoE methods* (2013) analizó la eficacia de la publicidad de video en Internet lineal utilizando métodos de calidad de experiencia. Otra tarea importante es la influencia de la entrega de video complementario utilizado en el proceso de enseñanza en la calidad de experiencia de los estudiantes.

Por lo tanto, la calidad de experiencia y los resultados de aprendizaje de los estudiantes puede depender del contexto de los videos complementarios insertados y del método para la incorporación de video en el material didáctico.

2.3.2 Videos Educativos

Se puede considerar material didáctico todo aquel que sea utilizado por el docente en función del logro de objetivos propuestos por él, de esta manera, un video educativo puede considerarse tanto el video que haya sido concebido con la finalidad puramente didáctica como aquel que no haya sido creado con ese fin, ya que puede resultar beneficiosa su utilización en el campo educativo, en este sentido es necesario la intervención más activa por parte del docente.

Tanto para los videos concebidos con un propósito didáctico como los que no, se han establecido sus funciones de la siguiente manera.

- **Informativa:** Estructurar la realidad.
- **Instructiva:** Orientar, condicionar el aprendizaje.
- **Motivadora:** Atraer, interesar, sensibilizar.
- **Evaluadora:** Auto observación, el análisis.
- **Investigadora:** Investigar, por ejemplo, a partir de grabaciones.
- **Metalingüística:** Del lenguaje audiovisual.
- **Expresiva:** Grabación, edición.
- **Otras funciones podrían ser:** lúdica, testimonial, innovadora.

Dado el bajo coste y el fácil manejo de los videos como recurso didáctico permiten estar presentes como un **Medio de Observación**, como **Medio de expresión**, como **Medio de Autoaprendizaje** y como **Medio de Ayuda a la Enseñanza** en diversos momentos del proceso educativo.

2.3.2.1 Medio de Observación

El video permite captar la realidad, la cámara es la encargada de registrar las situaciones que se están desarrollando, es la que recoge con mayor rigor y calidad lo que allí sucede, apoyándose en la idea de que el video registra realidades, es posible que nos sirva como una manera de formar y perfeccionar al profesorado, además de la observación de los alumnos, observación directa como medio de registro de datos en procesos de investigación y la observación de conductas con objetos de evaluación.

- **La formación y el perfeccionamiento del profesorado**

Es un medio excepcional para la auto-observación, análisis y corrección inmediata de la acción del profesor.

- **Observación de los alumnos**

Los profesores y tutores pueden emplear el vídeo como medio de estudio del comportamiento de los alumnos.

Al mismo tiempo los alumnos pueden observarse a través de la cámara y valorar su comportamiento. Esto ya es usado hoy en día por ejemplo en guarderías para transmitir a los padres el comportamiento de sus hijos en las aulas.

- **Observación directa como medio de registro de datos en proceso de investigación**

La cámara se convierte en un instrumento valioso en cuanto a la recogida de datos se refiere, es por tanto un medio de investigación importante. Los datos posteriormente serán examinados siguiendo determinadas metodologías y conservados para que el equipo investigador lo use a su disposición.

- **Observación de conductas con fines de evaluación**

La observación y auto-observación pueden ser utilizadas también como un medio de evaluación, tanto de los alumnos como de los profesores en período de formación.

2.3.2.2 Medio de Expresión

Al enfrentarnos a un video o programas debemos reaccionar ante ellos y no limitarnos a ser meros espectadores, existen dos tipos de objetivos que persigue la reacción antes los videos:

- Analizar los mensajes que transmiten los medios multimedia
- Los medios multimedia nos permiten experimentar una nueva "forma de expresión", estas formas de expresión son útiles tanto para los profesores como para los alumnos, por este motivo las distinguimos en dos apartados diferentes.
 - **Medio de expresión para los alumnos:** Los estudiantes pueden ayudar al profesor a confeccionar su programa de enseñanza, mediante opiniones o sugerencias que facilitarían el trabajo del profesor, por este motivo el Video es un medio de expresión y aprendizaje para el estudiante.
 - **Medio de expresión de los profesores:** existen dos puntos de vista por parte del profesor:
 - Uso de videos como un aficionado más de la narrativa, o como recurso meramente de información, e incluso como un medio de dotes artísticas.
 - Empleo de Videos como apoyo en presentación de trabajos académicos de investigación.

2.3.2.3 Medio de Autoaprendizaje

El video es ya utilizado por muchos de nosotros como un método de autoaprendizaje didáctico, mediante la transmisión de un video el alumno puede llegar a dominar los contenidos que se traten, la capacidad de dominar el contenido le puede servir como:

- Complemento Curricular: ampliamos los conocimientos para conseguir mejores trabajos, esta labor es conseguida gracias al autoaprendizaje.
- Autoenseñanza
- Enseñanza ocupacional
- Enseñanza a distancia
- Divulgación

Para el correcto aprendizaje, el video proporciona las interacciones con él, es decir, el video permite que podamos parar la imagen, dar marcha atrás, avanzar, en definitiva estas acciones permiten adecuarnos a las dificultades de comprensión o retención que podamos tener en el mismo momento de la reproducción del video.

2.3.2.4 Medio de Ayuda a la Enseñanza

Como ya hemos indicado anteriormente el video dentro de la enseñanza es un medio bastante utilizado dado que ayuda en el proceso enseñanza-aprendizaje, al ser una herramienta para transmitir información.

El video puede ser usado como complemento en una clase impartida por un docente, permitiendo que el video ilustre, más fácilmente los contenidos, ganando con esto en claridad y sencillez, convirtiendo así la clase más amena tanto para el alumno como para el profesor.

A continuación se explica como el video se puede convertir en un apoyo docente o como instrumento de transmisión de conocimientos:

- ***Vídeo-apoyo o complemento docente:*** Las imágenes, videos con o sin sonido, pueden ser un elemento que utilice el profesor para ilustrar su explicación. Nuestra cámara doméstica se puede convertir en una herramienta que nos ayude en la tarea de almacenamiento de video e imágenes que en su momento pudieron ser tomadas con el fin de entretener y que en momentos determinados nos puedan llegar a servir como elemento didáctico.
- ***Vídeo como instrumento de transmisión de conocimientos:*** El video puede servir de ayuda para el repaso de explicaciones repetitivas, así como en explicaciones de tipo matemático o incluso ejercer de estímulo, puesto que después de una explicación larga o de difícil comprensión puede ser un descanso su reproducción tanto para el profesor como para el alumno, siempre y cuando no interrumpa la actividad docente. Se indica a continuación algunas situaciones en la que el video puede ser instrumento de transmisión de conocimientos:
 - Grabación de lecciones impartidas por profesores o especialistas con prestigio.
 - Grabación de prácticas realizadas en un laboratorio: podemos pensar en la grabación del montaje o desmontaje de equipos de laboratorio, aunque no es un equipo de laboratorio en la red existen números videos que enseñan a montar dispositivos móviles, equipos portátiles, partes de un automóvil etc.
 - Documentales: información de aspecto científico, tecnológico o cultural. Antiguamente observar un documental era posible en la televisión, ahora es posible ver este tipo de videos en Internet. Hay que resaltar que son videos bien documentados, sacados de fuentes reales y con aportaciones de personas muy calificadas.
 - Recogen la información de centros de enseñanza: el video puede ayudar a los alumnos a conocer de antemano los centros en los que se han matriculado, presentándonos sus instalaciones, servicios que ofrecen, y actividades escolares o extraescolares, orientación escolar, etc.

2.4 Sistema de Aprendizaje mediante el uso de Videos

2.4.1 Introducción

Se tiene como motivación fundamental crear una "herramienta de ayuda didáctica para el aprendizaje de asignaturas mediante el uso de videos", basándonos en esta idea se crea una aplicación que permita hacer uso de los videos, de una forma algo distinta, no solo permitir la reproducción meramente del video, si no dar una opción un poco más allá, por esta razón se ha pensado en que el usuario pueda interactuar con el mismo, es decir proporciona al usuario un conocimiento de si el video seleccionado está siendo de ayuda o no a su aprendizaje, en el mismo momento de la reproducción. Para dar esta posibilidad al usuario es necesario establece un forma de evaluación, por ello se ha pensado en presentar una serie de preguntas tipo test que irán apareciendo automáticamente a medida que se reproduzca el video y a las cuales el usuario deberá ir contestando, permitiendo así repetir o no ciertos fragmentos del video dependiendo de si la pregunta ha sido contestada de forma correcta o incorrecta. De esta manera el usuario interactúa con el video y al mismo tiempo le permite conocer si está aprendiendo de él o no.

Es por tanto fundamental interactuar con la aplicación para reproducir por completo un video seleccionado. En este sentido es esencial que el usuario siga las siguientes operaciones que se repiten por cada interacción.

- Realizar una búsqueda, bien por subasignaturas o por un término general.
- Obtener una lista de videos como resultado de la búsqueda anterior.
- Acceder al video seleccionado para su reproducción.
- Una vez se reproduzca el video irán apareciendo automáticamente las preguntas a las cuales hay que responder eligiendo alguna respuesta de las proporcionadas
- Una vez elegida una respuesta, en el caso de ser contestada de forma correcta, se continua con la secuencia normal del video, y por el contrario si es contestada de forma incorrecta se repite el tramo del video que si lo visualiza con mucha atención encontrará la respuesta correcta.

2.4.2 Módulos que conforma el Sistema

2.4.2.1 Módulo de Registro de Videos didácticos en la aplicación

Se debe realizar un registro previo de los videos que conformaran la aplicación, para ello es necesario realizar los siguientes pasos:

- Registro de las Asignaturas y SubAsignaturas

Al tener que dar una clasificación a los videos, se ha pensado en hacerlo por asignaturas y subasignaturas, por este motivo es necesario establecer un listado a ser presentado a los usuarios para que se hagan una idea de los videos que hay a su disposición.

Para el registro tanto de asignaturas como de subasignaturas es necesario un nombre que las identifique unívocamente en la base de datos.

- Registro de Videos y Preguntas

Para el registro de Videos es necesario tener previamente registrado las asignaturas y subasignaturas y así establecer la clasificación del video.

Es importante registrar al mismo tiempo del video, las preguntas tipo Test que acompañaran al video una vez se reproduzca y aparezcan estas automáticamente en el momento de la reproducción.

Para registrar el video debemos registrar un título, descripción, asignatura y subasignaturas, además del nombre del video para su localización.

Para las preguntas es necesario identificar el video al que está asociado, además del enunciado de las preguntas y sus respuestas, es necesario indicar en el campo respuesta la solución correcta a dicha pregunta y en el campo orden, el tiempo en el que debe aparecer en el video.

2.4.2.2 Módulo Consulta en la Base de Datos

La forma de realizar consultas a la base de datos que proporciona nuestro sistema, es por dos campos: por una palabra clave o seleccionando del catálogo de subasignaturas.

Palabra Clave: en la página principal aparece un campo para ser rellenado por el usuario introduciendo un término general de búsqueda, por ejemplo: Aritmética, Química, Inglés, Ecuaciones Matemáticas, Matrices, etc. Una vez introducido el término de búsqueda se hace la consulta en la base de datos los videos que contengan ese término tanto en la descripción, o en título del video, o en la asignatura o en la subAsignatura. El sistema proporciona todos los resultados que haya localizado con esa palabra clave.

Catálogo de SubAsignaturas: el usuario tiene disponible una catálogo de subasignaturas que han sido registrados en la base de datos y de las que tendríamos registrados los videos, de este forma se facilita la búsqueda de los videos que le interesa al usuario ya que la búsqueda da como resultado solo los videos que están clasificados por una subAsignatura que pertenece a una asignatura determinada.

2.4.2.3 Módulo de Interacción con los Videos

Una vez se reproduce el video irá apareciendo automáticamente las preguntas según los tiempos que se haya fijado previamente cuando se registraron las preguntas, según las respuestas que se proporcionen en el caso de ser correcta el video continuara con la reproducción del video, y en caso de ser incorrecta repetirá la reproducción del video desde el segundo siguiente al que apareció la anterior pregunta y a la que se contesto de forma correcta.

2.4.2.4 Módulo de Colaboración con la Aplicación

Es muy importante disponer de videos, que no sean meramente videotutoriales, sino que hayan sido creados de tal manera que las preguntas tipo test encajen con cierta precisión y tengan sentido al interrumpir la reproducción normal del video, por este motivo se ha pensado crear un formulario de colaboración para las personas que desean ceder los videos sin ánimo de lucro y cuyo fin sea ayudar al estudiante. Este apartado es de gran valor para la aplicación ya que esto nos ayudaría hacernos con una amplia base de datos y así abarcar el máximo de asignaturas posibles y ser una verdadera ayuda para el estudiante.

2.4.3 HTML5 como soporte integrado para el contenido multimedia

Gracias a la evolución que ha tenido HTML hoy en día es posible hacer uso de la versión 5 para insertar contenido multimedia en documentos HTML. Por este motivo es usado en nuestro sistema de aprendizaje.

Antes de hablar de HTML5 vamos a tratar la evolución que ha tenido HTML a lo largo de los años.

2.4.3.1 Evolución de HTML



Figura 3. Icono de HTML

Tim Bernes-Lee, es el desarrollador original de HTML, en los años siguientes surge la necesidad de crear un estándar permitiendo de esta manera a los programadores y a los navegadores seguir las mismas normas para escribir HTML. La creación del estándar surge como consecuencias de los pasos agigantados en los que se ha evolucionado la Web.

Los estándares oficiales de HTML son: HTML 2.0, HTML 3.2, HTML 4.0, el HTML 4.01 y HTML 5. El HTML 5 es la última especificación oficial y se espera que continúe evolucionando a lo largo de los próximos años.

La siguiente tabla resume la evolución de HTML:

HTML 2.0	<p>En 1995 se publica el estándar HTML 2.0. A pesar de su nombre, HTML 2.0 es el primer estándar oficial de HTML, es decir, el HTML 1.0 no existió como estándar. HTML 2.0 no soportaba tablas.</p> <p>Se simplificaba al máximo la estructura del documento para agilizar su edición, donde la declaración explícita de los elementos body, html y head es opcional.</p>
-----------------	---

HTML 3.2	La versión HTML 3.2 se publicó en 1997 y es la primera recomendación de HTML publicada por el W3C (Consortio internacional). Esta revisión incorporó los últimos avances de las páginas web desarrolladas hasta 1996, como applets de Java y texto que fluye alrededor de las imágenes.
HTML 4.01	La última especificación oficial de HTML se publicó en diciembre de 1999 y se denomina HTML 4.01. Desde la publicación de HTML 4.01, el W3C se centró en el desarrollo del estándar XHTML. Por este motivo, en el año 2004, las empresas Apple, Mozilla y Opera mostraron su preocupación por la falta de interés del W3C en HTML y decidieron organizarse en una nueva asociación llamada WHATWG (Web Hypertext Application Technology Working Group) que comenzó el desarrollo del HTML 5, cuyo primer borrador oficial se publicó en enero de 2008. Debido a la fuerza de las empresas que forman el grupo WHATWG y a la publicación de los borradores de HTML 5.0, en marzo de 2007 el W3C decidió retomar la actividad estandarizadora de HTML, dentro del cual decidió integrar el XHTML.
HTML 5, HTML 5.1, HTML 5.2	El consorcio internacional W3C, después de una evolución de varios años, liberó el HTML 5 como estándar oficial a finales de octubre de 2014. HTML 5 incorpora nuevos elementos no contemplados en HTML 4.01. Hay diversos cambios respecto a HTML 4.01. Hay nuevas etiquetas, se introduce la posibilidad de introducir audio y video de forma directa en la web sin necesidad de plugins o complementos en los navegadores, y otras novedades. El W3C irá lanzando progresivamente nuevas evoluciones del HTML 5.

Tabla 5. Evolución de HTML

Hasta hace poco la versión de HTML más utilizada ha venido siendo la 4.01, que es admitida por todos los navegadores. Esta versión fue definida por la W3C (Comité Internacional que define los estándares web) hace varios años. Actualmente ya está utilizándose la nueva versión de HTML, denominada HTML 5, aunque esta no es soportada por navegadores antiguos. HTML 5 se ha ido introduciendo progresivamente en los desarrollos web pero aún quedan muchos años en que existirá una “convivencia” con navegadores y webs que usan HTML 4.01.

2.4.3.2 HTML5



Ilustración 4. HTML5

- Una primera diferencia destacable de HTML5, es que desaparecen las variantes Strict, Transitional y Frameset, muy presentes en HTML4.
- La cualidad más importante y destacable de esta nueva versión de HTML está relacionada con la inserción multimedia en los sitios webs, para poder ser incluidos se contará con etiquetas HTML.

El objetivo que se persigue con insertar recursos multimedia mediante etiquetas HTML es que los usuarios puedan consumir los videos o canciones sin necesidad de instalar nada de forma adicional.

La sintaxis de HTML5 se destaca en el ámbito multimedia, pero se han introducido bastantes etiquetas nuevas para generar una mejoría y que el proceso de escribir código sea más simple y lógico.

A continuación indicaremos las etiquetas HTML que nos permite la labor del uso de recurso multimedia, así como otro tipo de etiquetas que se ha incluido en esta nueva versión.

- **audio y video:** estas son las dos más importantes etiquetas de HTML5, dado que nos permiten acceder de forma más simple a contenido multimedia que puede ser reproducido por casi todo tipo de dispositivos, marcan el tipo de contenido que estará en su interior.
- **article:** esta etiqueta sirve para definir un artículo, un comentario de usuario o una publicación independiente dentro del sitio.
- **header, footer:** estas etiquetas individuales ahorran tener que insertar IDs para cada uno, como se solía hacer anteriormente. Además, se pueden insertar headers y footers para cada sección, en lugar de tener que hacerlo únicamente en general.
- **nav:** la negación puede ser insertada directamente en el markup, entre estas etiquetas, que nos permitirán hacer que nuestras listas oficien de navegación.
- **section:** con esta etiqueta, una de las más importantes de las novedades, se puede definir todo tipo de secciones dentro de un documento. Por ponerlo de forma sencilla, funciona de una forma similar a la etiqueta div que nos separa también diferentes secciones.

- **embed:** con esta etiqueta se puede marcar la presencia de un contenido interactivo o aplicación externa.
- **canvas:** finalmente, esta etiqueta nos permite introducir un “lienzo” dentro de un documento, para poder dibujar gráficos por vectores; será necesario el uso de JavaScript.
- Usar HTML5 puede reducir la dependencia de los plugins, como ocurre en el caso de Adobe Flash que depende directamente de un plugins para reproducir los videos o el sonido en la Web, por este motivo con los avance conseguidos por HTML5 se ve claramente perjudicado, ya que mediante este golpe proporcionado por HTML5 cada vez se está usando menos.
- Con los avances conseguidos en HTML5 es posible que el usuario pueda acceder a sitios web sin estar conectados a internet, para llevar a cabo este hecho se va almacenando localmente en el navegador las web que lo implementan.
- Mejora el renderizado de mapas y lo sistemas de geoposición del navegador, por tanto la geolocalización es otro de sus puntos fuertes.
- Funcionalidad "drag and drop".
- Edición online de contenidos, ampliamente popularizada por Google Docs.

En definitiva:

HTML5 es la última versión de HTML. El término representa dos conceptos diferentes:

- Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos.
- Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance.

Las funcionalidades que ofrece HTML se resumen de la siguiente manera:

- *Semántica:* Permite describir con mayor precisión cuál es su contenido.
- *Conectividad:* Permite comunicarse con el servidor de formas nuevas e innovadoras.
- *Sin conexión y almacenamiento:* Permite a las páginas web almacenar datos localmente en el lado del cliente y operar sin conexión de manera más eficiente.
- *Multimedia:* Nos otorga un excelente soporte para utilizar contenido multimedia como lo son audio y video nativamente.
- *Gráficos y efectos 2D/3D:* Proporciona una amplia gama de nuevas características que se ocupan de los gráficos en la web como lo son canvas 2D, WebGL, SVG, etc.
- *Rendimiento e Integración:* Proporciona una mayor optimización de la velocidad y un mejor uso del hardware.
- *Acceso al dispositivo:* Proporciona APIs para el uso de varios componentes internos de entrada y salida de nuestro dispositivo.

2.4.4 Aplicaciones con Videos Educativos

A continuación se describen varias Web que hacen uso de Videos como método didáctico de autoaprendizaje.

Youtube

El canal de videos se ha convertido en una herramienta fundamental a la hora de reforzar contenidos que quizá no se comprendieron en clases o para aprender cosas nuevas que complementen un saber previo. Basta escribir una palabra, materia o tema para acceder a miles de tutoriales educativos. Una muestra más de que son muchos los docentes que implementan en sus clases las herramientas que las nuevas tecnologías ponen a su disposición, por este motivo los profesores de todo el mundo generan videos didácticos y posteriormente lo suben a esta plataforma para que los alumnos, y otros docentes, puedan verlos y consultarlos.

Mediateca

Nacen como centros de recursos diseñados para atender a las necesidades culturales educativas o de investigación de la comunidad académica. La mediateca es la sección formada con los medios de información no librarios, especialmente audiovisuales, de la biblioteca, donde se seleccionan, reúnen, ordenan y se ponen a su disposición de los usuarios. Cuenta con personal especializado que asesora y forma a sus usuarios en el manejo de las herramientas de acceso y tratamiento de la información: Consulta a bases de datos, edición y tratamiento de textos, imagen, sonido y videos.

Proyecto Biosfera

Recoge una relación de vídeos de Biología y Geología, recopilados por el Taller de Informática Educativa del IES Arencibia, como un complemento didáctico para el aula. Los vídeos se completan con ejercicios para valorar el aprendizaje del estudiante tras su visionado. En la misma página web del proyecto se recogen múltiples enlaces a otros vídeos relacionados con estas áreas curriculares.

Video digital educativo

Una colección de vídeos sobre distintos contenidos tratados en clase, realizados por los alumnos del colegio Erain Ikastetxea School de Gipuzkoa. Están clasificados por categorías como antropología, expresión oral, música, geología o educación en valores.

Educatube

Recopila un buen número de vídeos educativos para primaria, secundaria y bachillerato organizados por materias curriculares. Como complemento, cada vídeo cuenta con una descripción de su contenido.

Videoteca Educativa

Este recurso de la consejería de Educación de Murcia recopila un amplio grupo de vídeos didácticos clasificados por niveles educativos y materias. En cada vídeo, se hace referencia al contenido y a su duración.

Videoteca de Cine

Un espacio para docentes y educadores que deseen utilizar y compartir experiencias y materiales que sirvan para utilizar el cine como herramienta didáctica en las aulas. Los vídeos están clasificados por áreas curriculares y etapas educativas o por formato y técnica utilizada.

Aula Visual

Dirigida a los docentes con el fin de motivar el uso de los medios audiovisuales como recursos de apoyo para el aula. Permite ver, descargar y subir vídeos y su catálogo está conformado por material producido por la industria audiovisual, por profesores para sus clases y por alumnos, como actividad de una asignatura.

TeacherTube

Servicio que vio la luz en Internet en el 2007. Su filosofía es similar a la del famoso Youtube, pero con la finalidad de estar centrado con la enseñanza. Se pueden buscar en su base de datos videos, archivos de audio, fotos, y documentos en diversos formatos relacionados con un tema educativo simplemente escribiendo en el buscador las palabras claves adecuadas.

CiberMatex

Sistema de Aprendizaje de las Matemáticas basados en el video-explicación. Miles de videos tipo "youtube" son accesibles a sus suscriptores. Las explicaciones pausadas y directas aseguran una fácil comprensión de los contenidos, incluso para quienes retoman sus estudios después de varios años.

Educatina

Plataforma de educación digital que ofrece miles de videos y prácticas interactivas totalmente gratuitas para utilizarlas como complemento del material que se trabaja en las aulas. Ofrece explicaciones animadas sobre los contenidos sobre los contenidos que se dictan en el aula y diversos contenidos adicionales de interés general. El desarrollo del material está a cargo de docentes especializados en cada área temática y es supervisado por profesionales para garantizar la calidad audiovisual y el contenido académico. De esta manera, los alumnos tienen acceso a presentaciones didácticas sencillas que les ayudan a entender y aprender los conocimientos trabajados durante la jornada escolar.

2.5 Ventajas que ofrecen los videos

Las ventajas del vídeo como recurso didáctico son múltiples:

- La retención de contenidos es mayor.

RETENCIÓN DE CONTENIDOS	
10 %	De lo que oímos
20 %	De lo que vemos
40 %	De lo que vemos y oímos
75 %	De lo que vemos oímos y hacemos

Figura 5. Retención de Contenidos

- Ofrecen la posibilidad de “trasladarse” a espacios y tiempos diversos.
- Desarrollan la sensibilidad. Conlleva un estímulo afectivo (condiciones de aprendizaje más rápidas y duraderas).
- Contribuyen a la mejor comprensión de la temporalidad, las relaciones causa-efecto y las explicaciones multicausales de los hechos humanos.
- Contribuyen a que el alumnado supere el pensamiento concreto, egocéntrico, y acceda al pensamiento formal, abstracto.
- Permiten la atención de alumnos con bajo nivel de interés.
- Posibilita la interacción en el aula (trabajo colectivo).
- Integrador de otros medios audiovisuales (cine, fotografía, etc.).
- Readaptador de otros recursos (cine, fotografía, entorno, otros medios).
- Fácil manejo y relativo bajo costo.
- Permite manipular con facilidad la cinta u otro soporte para adaptarlas a las necesidades.
- Permite duplicar, catalogar, almacenar, conservar y modificar la información.

CAPÍTULO 3

3. Tecnología y Herramientas

En este capítulo se describen las diferentes tecnologías empleadas durante el proceso de desarrollo de la aplicación.

3.1 Lenguajes de programación

Los lenguajes de programación son un conjunto de instrucciones consecutivas que se ejecutan en un procesador. Las instrucciones que ejecuta el procesador están escritas en lenguaje máquina, es decir, en lenguaje binario, incomprensible para el ser humano, razón por la que se crearon lenguajes comprensibles. Se comenzó con la creación del lenguaje ensamblador, muy parecido al lenguaje máquina pero entendible. Este lenguaje tiene un gran problema y es que depende del procesador, ya que cada procesador puede tener su propio lenguaje máquina.

Los lenguajes de programación se pueden clasificar en:

Compilados: lenguaje que para ejecutarse es necesario producir un archivo ejecutable, a través de un compilador. El típico ejemplo de lenguaje compilado es C.

Interpretados: lenguaje que debe ser traducido al lenguaje máquina, a través de un programa auxiliar o interprete para ser ejecutado. Un ejemplo de lenguaje interpretado puede ser PHP.

Existen otros lenguajes que son tanto interpretados como compilados, como es el caso de Java.

3.1.1 Java

Java es un lenguaje de programación creado por Sun Microsystems en 1995. Este lenguaje es uno de los más extendidos en el desarrollo de aplicaciones, juegos y herramientas. Además es una plataforma informática y está presente en miles de millones de dispositivos móviles y ordenadores.

3.1.2 Lenguajes Dinámicos

Los lenguajes dinámicos son aquellos lenguajes que permiten la modificación del código en tiempo de ejecución y que no requieren tiempo de compilación. En la actualidad existen muchos lenguajes dinámicos, como pueden ser PHP, Python, Ruby o Groovy, pero nos centraremos en este último por estar basado en Java.

3.1.2.1 Groovy

Groovy es un lenguaje de programación dinámico que se ejecuta sobre la máquina virtual de Java y creado por SpringSource. Este lenguaje está influenciado por otros lenguajes dinámicos como Python, Ruby o Smaltalk, pero basado principalmente en Java. Entre las principales características, destacan el tirado dinámico, el uso de *closures* (bloques de código que se asignan a una variable y se ejecutan a posteriori), soporte para expresiones regulares y la sobrecarga de operadores.

En la siguiente imagen se puede observar un ejemplo de una *closure*. En ella se itera sobre cada elemento de la lista para calcular el sumatorio.

```
def method = {
  def list = [4, 8, 15, 16, 23, 42]
  def sum
  list.each { element ->
    sum += element
  }
}
```

Figura 6. Closures de Groovy

La síntesis de Groovy es sencilla y muy similar a Java, con algunas diferencias que hacen que el lenguaje sea mucho más legible. Estos cambios son:

- No es necesario poner punto y coma después de escribir una sentencia.
- No son necesarios los paréntesis.
- Los métodos de acceso a las variables, comúnmente denominados *getter* y *setter* no aparecen, pero vienen implícitos. Si es necesario añadir funcionalidad a esos métodos, basta con escribirlos con la nueva lógica.
- Los métodos que reciben parámetros pueden ser llamados con un menor número de parámetros. En el ejemplo se observa como el número de parámetros disminuye, incluso es posible invocar el método sin declarar ninguno de ellos:

```
def method(x=1,y=2) {
  x + y
}

method() //x + y = 3
method(2) //x + y = 4
method(2,3) //x + y = 5
```

Figura 7. Métodos en Groovy con varios parámetros

3.1.3 HTML5

Hypert Text Markup Language, versión 5 es la quinta revisión importante del lenguaje básico de la World Wide Web HTML. HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas <div> y , pero tienen un significado semántico, como por ejemplo <nav> (bloque de navegación del sitio web) y <footer>. Otros elementos proporcionan nuevas funcionalidades a través de una interfaz estandarizada, como los elementos <audio> y <video>. Mejora el elemento <canvas>, capaz de renderizar elementos 3D en los navegadores más importantes (Firefox, Chrome, Opera, Safari e Internet Explorer).

```

<!DOCTYPE html>
<html>
<body>
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogv" type="video/ogg">
  Your browser does not support the video tag.
</video>
</body>
</html>

```

Figura 8. Uso de HTML5 etiqueta Video

3.1.4 JavaScript

Javascript es un lenguaje de programación utilizado en páginas web, que el navegador es quien se encarga de ejecutarlo. Javascript dispone de muchas librerías que permiten ampliar funcionalidad y aplicar nuevos estilos en las aplicaciones web. Una de las librerías más utilizadas es JQuery.

JQuery es una librería Javascript que ofrece un mecanismo fácil para interactuar y modificar documentos HTML, crear efectos gráficos, manejar eventos e interaccionar con otras páginas web mediante tecnologías Ajax.

Con el uso de esta tecnología, se consiguen ahorrar muchas líneas de código para conseguir el mismo objetivo en el caso de no utilizar jQuery. Por ejemplo, en la Figura 9, se ejecuta una función, una vez ha cargado la página y almacena el valor de un campo cuyo identificador es "username".

```

$(document).ready(function(){
  var inputUsername = $("#username");
});

```

Figura 9. Uso de JQuery

- **jQuery-ui:** es la librería oficial de jQuery para crear interfaces. Con ella podemos definir diferentes estilos para los botones, pestañas, calendarios y otros elementos visuales de HTML. En la siguiente imagen podemos observar los diferentes estilos que se pueden aplicar a botones, pestañas, calendarios, campos de texto con autocompletado y barra de desplazamiento.

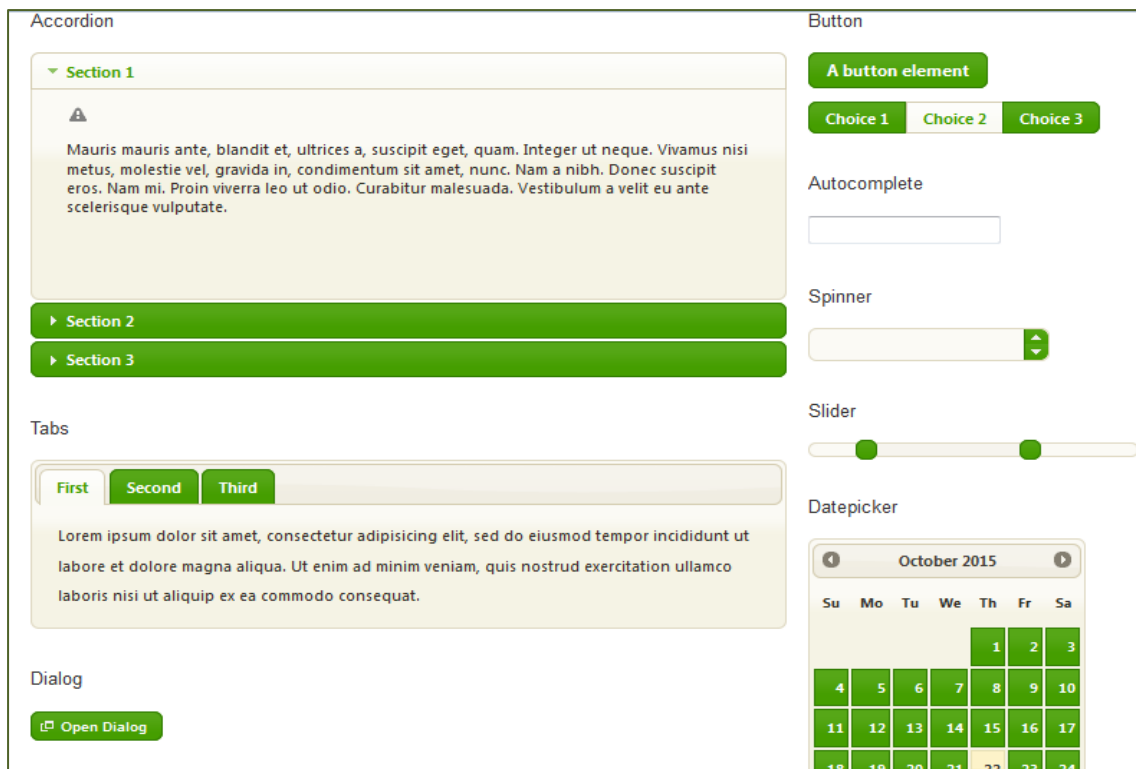


Ilustración 10. Uso de JQuery-UI

- **jQuery-DataTables:** plugins que nos permite darle dinamismo a nuestras tablas de una forma sencilla y aumentando vistosidad, además podremos conseguir las siguientes acciones.
 - Ordenar por Campos
 - Pagarinar Resultados
 - Filtrar
 - Sistema de Búsqueda
 - Obtención de datos de forma dinámica

Show entries

Search:

Name	Category	Location	From	
Arenas Doradas	★★★★★	Varadero	31.17 GBP	Rooms Pics Options Delete
Barcelo Marina Palace	★★★★★	Varadero	52.16 GBP	Rooms Pics Options Delete
Barcelo Solymar	★★★★★	Varadero	41.98 GBP	Rooms Pics Options Delete
Be Live Turquesa	★★★★★	Varadero	24.17 GBP	Rooms Pics Options Delete
Blau Varadero	★★★★★	Varadero	16.54 GBP	Rooms Pics Options Delete
Coralia Cuatro Palmas	★★★★★	Varadero	25.45 GBP	Rooms Pics Options Delete
Iberostar Laguna Azul	★★★★★	Varadero	36.26 GBP	Rooms Pics Options Delete
Iberostar Playa Alameda	★★★★★	Varadero	33.71 GBP	Rooms Pics Options Delete
Iberostar Tainos	★★★★★	Varadero	27.35 GBP	Rooms Pics Options Delete
Iberostar Varadero	★★★★★	Varadero	40.71 GBP	Rooms Pics Options Delete

Showing 1 to 10 of 19 entries (from 96 entries)

First Previous 1 2 Next Last

Figura 11. Uso de JQuery-DataTables

3.2 Framework de desarrollo

Un *framework* es una herramienta o conjunto de herramientas que permiten el desarrollo de una aplicación dentro de un determinado ámbito y no solo para el desarrollo de aplicaciones web. Podemos encontrar *frameworks* para el desarrollo de juegos o para crear aplicaciones de escritorio.

Los *frameworks* permiten ofrecer a los desarrolladores, no solo las herramientas con las que crear la aplicación, sino también que el desarrollo sea más rápido y promover el uso de buenas prácticas de programación.

Existen varios *frameworks* para el desarrollo de aplicaciones con el lenguaje Groovy. Entre ellos podemos encontrar:

- Grails.
- Griffon.
- WOGroovy.
- Aerie.

3.2.1 Grails

Grails es el *framework* para el desarrollo de aplicaciones web por medio del lenguaje Groovy. Dentro de las diferentes posibilidades para la creación de aplicaciones con Groovy, Grails es de los *frameworks* más utilizados por diferentes razones:

- Se ejecuta sobre la máquina virtual de Java (JVM).
- Utiliza la estructura del modelo vista controlador (MVC). En esta estructura, el modelo representa los datos o información almacenada, la vista donde se muestra

la información y el controlador, los mecanismos encargados de interactuar entre la vista y el modelo.

- Sigue varias filosofías:
 - *Don't Repeat Yourself* (DRY): no te repitas a ti mismo. Quiere decir que no vuelvas a escribir código ya escrito, principalmente utilizando *plugins*.
 - *Convention over Configuration*. Con Grails se intenta evitar que el programador dedique demasiado tiempo a los ficheros de configuración en XML.
- Ofrecer la creación de la representación de los datos (las vistas), mediante Grails Server Pages (GSP). Las páginas GSP proporcionan las mismas funcionalidades que las Java Server Pages, y además agregan nuevas funcionalidades que permiten mostrar contenido utilizando menos líneas de código. Utiliza *Grails Object Relational Mapping* (GORM). GORM es la implementación Grails de un mapeo relacional de objetos (ORM). Los ORM permiten establecer una relación entre el modelo de datos y la información almacenada en la base de datos, lo que permite operar sobre la información de manera sencilla. Además con GORM ya contamos con métodos implementados implícitamente como pueden ser *delete*, *save* o *update*
- No es necesario implementar métodos para realizar búsquedas en Grails se puede utilizar buscadores dinámicos, es decir si una clase de dominio "Asignaturas " tiene como atributo "nombre" podemos buscarlo en la tabla asignaturas invocando al método "Asignaturas.findByNombre(nombre)"

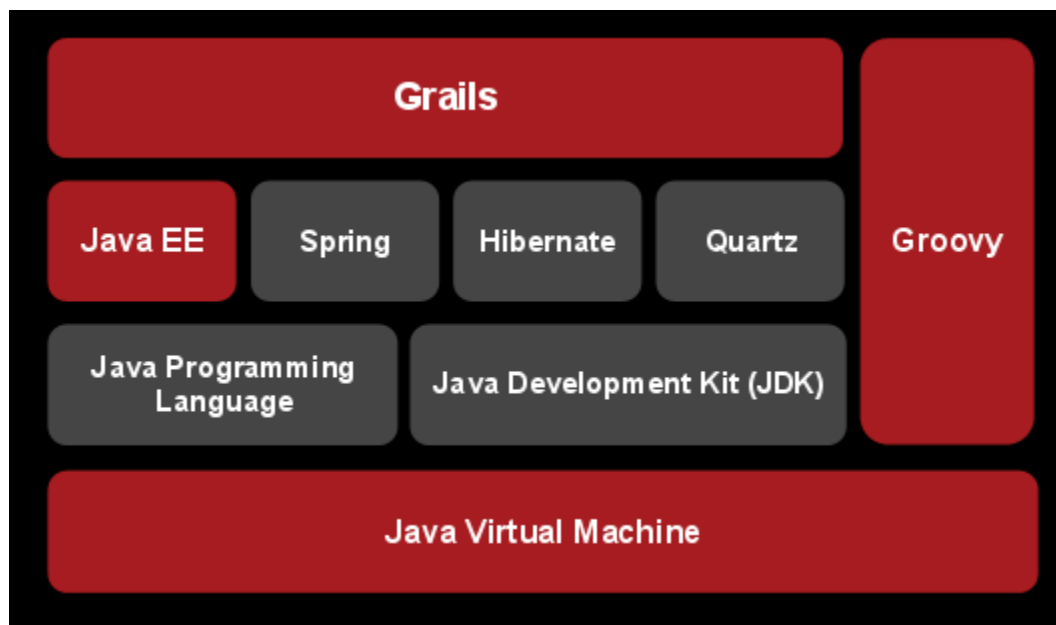


Figura 12. Arquitectura Grails

3.3 Plugins

Un *plugins* es un complemento software o hardware que permite ampliar una característica o agrega un servicio a un sistema más grande.

En el caso de Grails, los *plugins* son paquetes que permiten agregar funcionalidades a nuestro proyecto.

3.3.1 jQuery

Plugins que ofrece la integración de la librería Javascript jQuery, anteriormente mencionada, con Grails. Esto nos permite tener un control sobre las extensiones que utilizamos, permitir acceder de forma más cómoda a los recursos que ofrece jQuery y conocer la versión con la que se trabaja.

3.3.2 jQuery-ui

Al igual que con el *plugins* de jQuery para Grails, con este conseguimos evitar duplicidades y conflictos, manteniendo de forma integrada nuestro *framework* con las diferentes extensiones que este nos permite utilizar.

3.3.3 Mail

Mail permite el envío de email desde la aplicación de una manera sencilla. Podemos enviar simple correos con un texto o paginas gsp con el contenido que nos interesa. Para su uso hay que realizar una serie de configuraciones en los archivos Conf del entorno, esto viene documentado en la página de Grails.

3.3.4 Simple Captcha

Plugins que permite incluir en nuestras vistas imágenes captcha como control de seguridad para envíos de datos obtenidos por el uso formularios en la aplicación. Es necesario incluir ciertas configuraciones en los archivos Conf del entorno, al igual que el plugins Mail, viene documentado en la página de Grails.

3.4 Entornos de desarrollo integrados (IDE)

Un entorno de desarrollo integrado es un elemento software, que incorpora diferentes herramientas que permiten al programador crear código. Los IDE pueden estar contruidos para un solo lenguaje, como por ejemplo JBuilder que se utiliza para desarrollar en el lenguaje Java, o para varios lenguajes como puede ser Eclipse, que da soporte para desarrollar, principalmente en Java, pero también para C, C++, Cobol, PHP, etcétera.

3.4.1 SpringSource tool suite

SpringSource Tool Suite (STS) es un IDE basado en Eclipse y extensible mediante *plugins* creado por SpringSource. Este IDE está disponible para múltiples plataformas, como Windows, Linux, Mac OS X. STS está orientado al desarrollo de Java, Spring y Groovy y Grails, siento este último el que utilizaremos. STS trae integrado vFabric tc Server Developer Edition, un servidor, mejor que Apache Tomcat, que también es de código abierto. Este servidor ofrece mejoras como la visualización del rendimiento de las aplicaciones, integración con STS, funciones mejoradas de diagnostico de problemas, etcétera.

También existen otros IDE que ofrecen soporte a Groovy y Grails, como por ejemplo, NetBeans o IntelliJ IDEA.

Se ha escogido STS como IDE para el desarrollo de este proyecto, frente a los otros IDE sobre los que se podría realizar el desarrollo, porque tanto Groovy como Grails son productos de SpringSource y STS está creado para potenciar las posibilidades que ofrecen el lenguaje y el *framework* elegidos

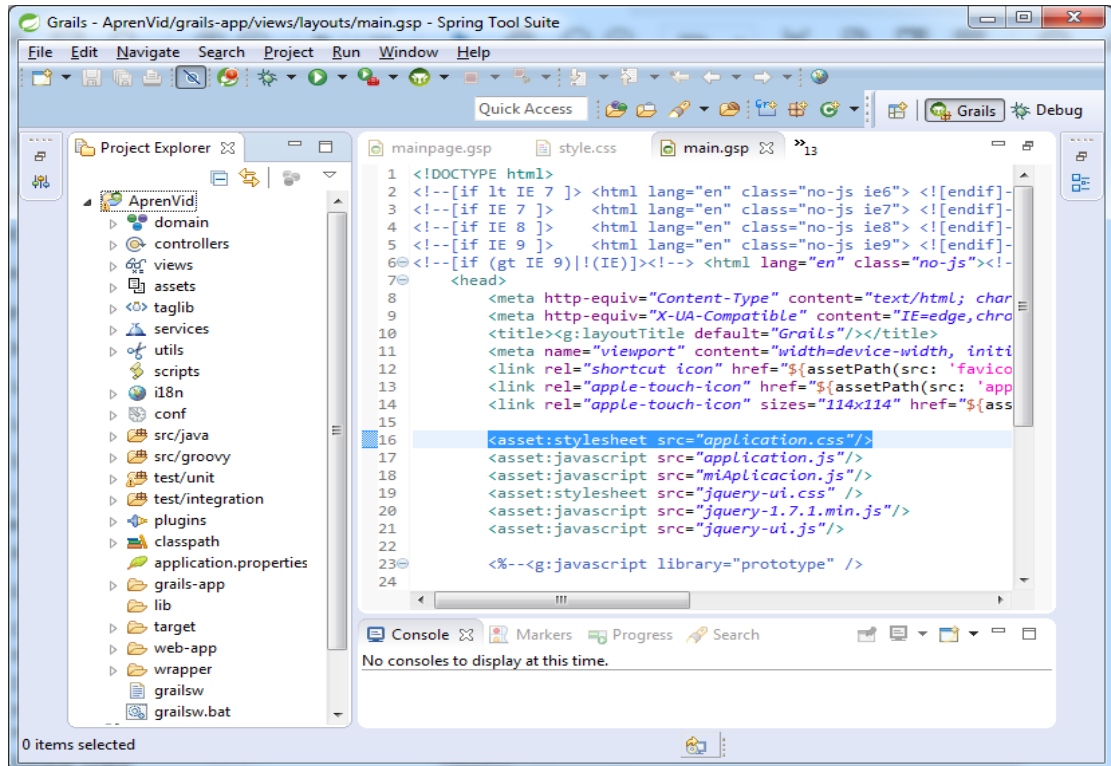


Ilustración 13. SpringSource Tool Suite

3.5 Otras herramientas

En este apartado se van a especificar las diferentes herramientas empleadas durante el desarrollo del proyecto.

3.5.1 Firebug

Es una herramienta que se integra con el navegador y permiten visualizar código HTML de una página web, los estilos de cada elemento HTML, editar estos contenidos dinámicamente y ver el resultado permitiendo capturar todas las llamadas que se realizan una vez la página está cargada.

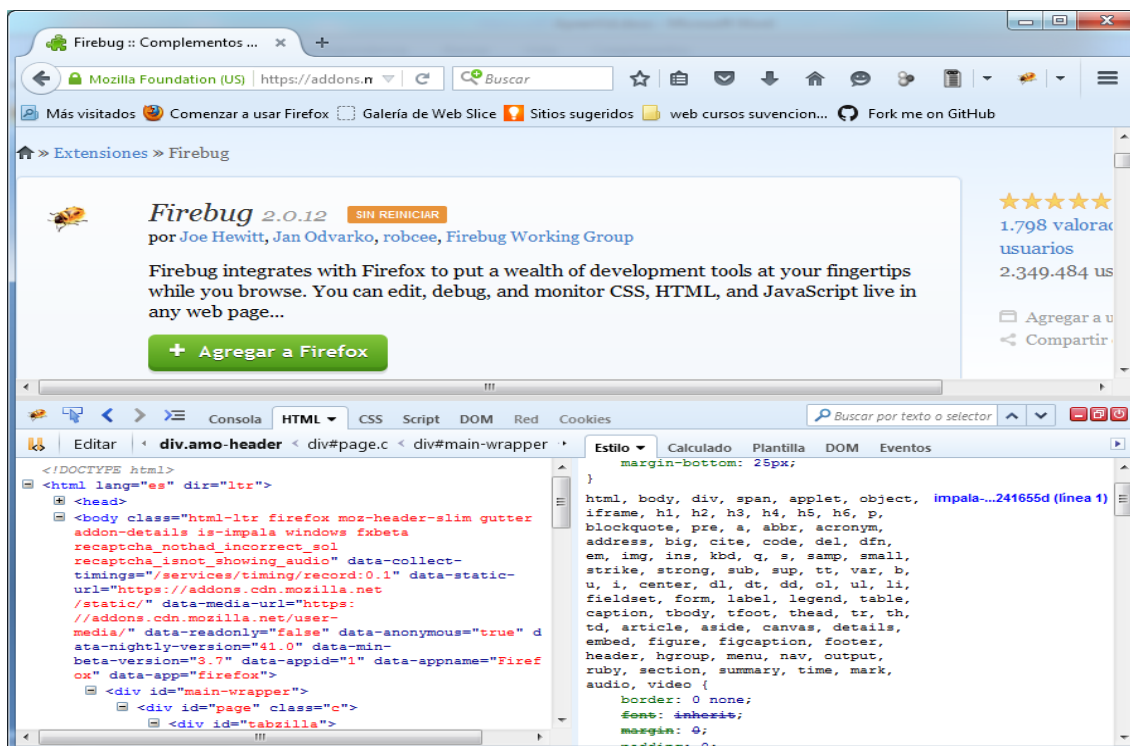


Figura 14. Firebug con código HTML y CSS

Esta herramienta ha sido de gran ayuda para depurar el código javascript

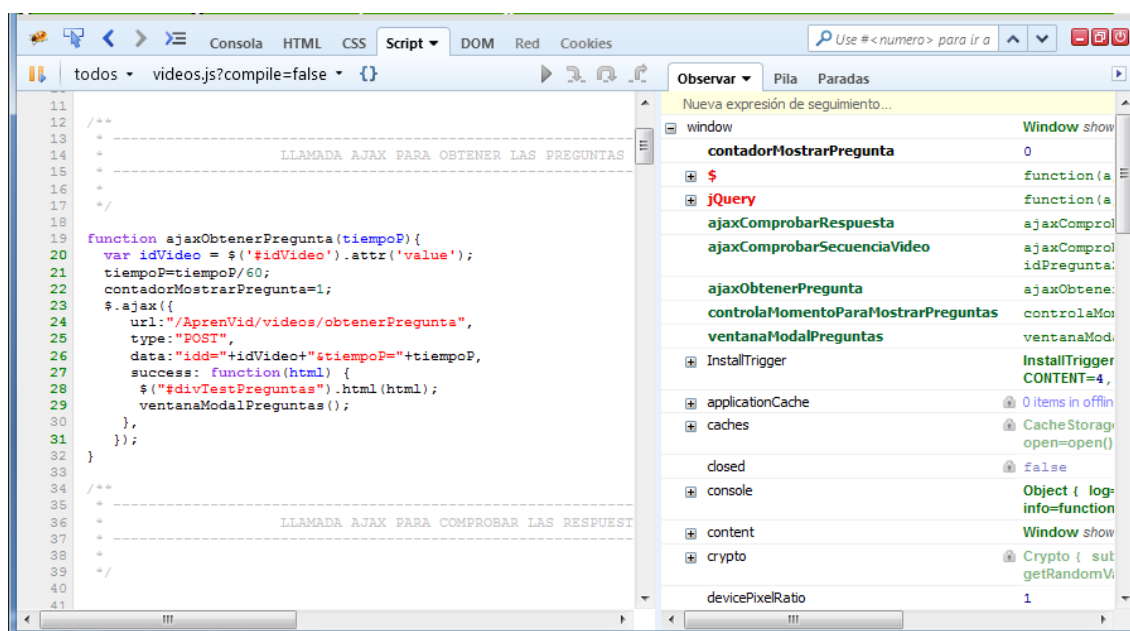


Figura 15. Firebug como depurador de JavaScript

3.5.2 Video DownloadHelper

Complemento del navegador Firefox que permite la descarga de videos, para el uso de este complemento es necesario acceder por ejemplo a Youtube y visualizar el video a descargar, en ese momento es cuando aparece el icono enmarcado en rojo, que se observa en la Figura16, al pinchar aparece una ventana con el programa Video DownloadHelper, seleccionamos cualquiera de los botones con el ícono más y en ese momento aparece la opción descargar.



Figura 16. Video DownloadHelper

3.5.3 Gimp 2

Programa de edición de imágenes digitales, es libre y gratuito Forma parte del proyecto GNU y está disponible bajo la Licencia pública general de GNU y GNU Lesser General Public License, gracias a este programa hemos podido crear el logotipo de nuestra aplicación.

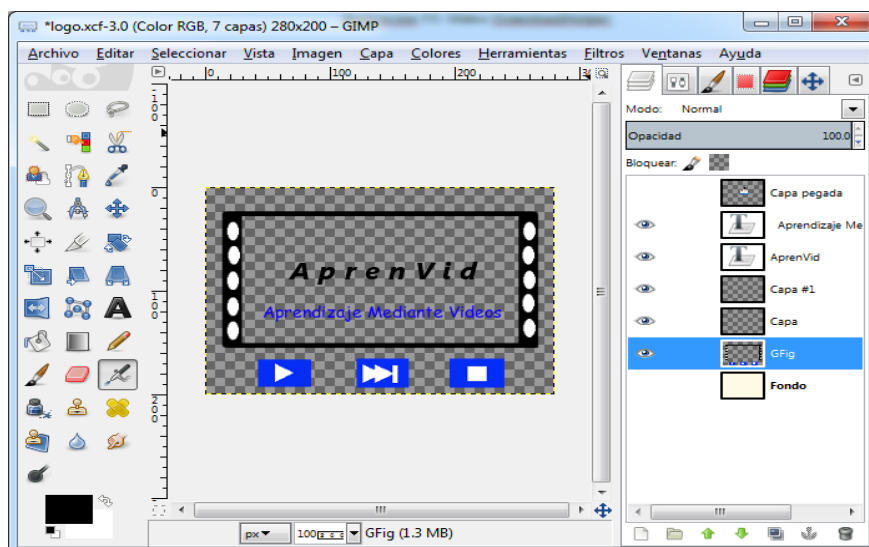


Ilustración 17. Uso de Gimp2

CAPÍTULO 4

4. Análisis del Sistema

En este capítulo se detallan los diferentes requisitos software que demanda la aplicación.

4.1 Identificación de escenarios y usuarios

El desarrollo de la aplicación está dirigido principalmente al sector estudiantil ya que son ellos quienes más buscan ayuda en las web sobre temas tratados en clase que no han sido asimilados de una forma correctamente o incluso aún siéndolo necesitan ser reforzados y que mejor forma de atraer su atención permitiéndoles interactuar con los elementos que son más utilizados en la tarea del aprendizaje, los Videos.

Por tanto existen dos escenarios posibles para la aplicación:

1. Uno de los escenarios corresponde a quienes usan directamente la aplicación mediante el empleo de videos con el propósito académico, es el caso de los estudiantes, y los administradores mediante el registro de los mismos hacen que estén disponibles para cualquier usuario que necesitan aprender, ya sean estudiantes o no.
2. Otro escenario posible corresponden a las personas que ceden los videos con las respectivas preguntas, ellos no utilizan directamente la aplicación, si no que su misión es tan solo rellenar un formulario con sus datos personales, esto datos serán enviados a los administradores de la aplicación, quienes son los que se pondrán en contacto con dichas personas y los que indicarán la forma de hacernos llegar tales videos. La colaboración juega un papel importante sobre todo para los inicios de la aplicación.

4.1.1 Usuarios

Para el uso de la aplicación podemos distinguir dos tipos de Usuarios:

- **Cliente:** en este caso los estudiantes o cualquier persona que desea instruirse, ellos son quienes hacen uso de los videos disponibles en la aplicación para su aprendizaje.
- **Administrador:** gestionan toda la información correspondiente a los videos.

Se detallan, a continuación, las diferentes acciones que pueden realizar cada uno de los usuarios.

4.1.1.1 Clientes

Las acciones de los clientes se centra principalmente en la consulta de videos de aprendizaje, las consultas se pueden realizar por dos tipos diferentes de búsquedas:

- **Búsqueda General:** mediante una palabra clave, el sistema hace una búsqueda de todos los videos que contengan esa palabra clave tanto en el título o descripción del video, así como en la asignatura o subAsignatura a la que este asociado el video.
- **Búsqueda por Asignaturas:** para minimizar el trabajo de búsqueda al usuario, se ofrece un catálogo que clasifica los videos por asignaturas y subasignaturas registradas en el sistema.

4.1.1.2 Administrador

El administrador es el responsable de hacer llegar al usuario el video con sus respectivas preguntas que permiten la interacción con el video en el mismo momento de su reproducción y por tanto mediante esta tarea ayudar en el aprendizaje.

El administrador es el encargado de realizar las siguientes acciones:

- Gestionar las asignaturas por la cuales deben ser clasificadas los videos en el momento de subirlos a la aplicación.
- Gestionar las subasignaturas que pertenecen a cada asignatura para ofrecer a los usuarios una mejor clasificación de la información.
- Gestionar los videos a los cuales van acceder los usuarios permitiendo que estos sean visualizados con un título y una descripción más detallada de la información que contienen.
- Gestionar las preguntas que aparecerán en los respectivos videos y así permitir al usuario interactuar con ellos, principal objetivo que se persigue con el desarrollo de este proyecto.

4.2 Establecimientos de requisitos del software

Se procede en este apartado a detallar el conjunto de requisitos necesarios para el desarrollo de la aplicación y se detalla la nomenclatura que se empleará.

Los requisitos estarán establecidos en Funcionales y No Funcionales, es necesario establecerlos de esta manera ya que permiten una mejor comprensión del alcance de la aplicación.

Como se ha indicado en el párrafo anterior los requisitos pueden ser:

- Funcionales: mediante estos requisitos permiten determinar la funcionalidad de la aplicación identificando aquellas acciones que pueden realizar los diferentes roles de la aplicación.
- No Funcionales: son los que determinan las restricciones o condiciones en aspectos de calidad, estándares, costes, estabilidad y portabilidad.

Para describir los requisitos tanto funcionales como no funcionales se preparan las siguientes tablas que recogen la información:

Identificador	
Título	<Título del Requisito>
Prioridad	<Prioridad>
Entrada	<Entrada>
Salida	<Salida>
Descripción	<Descripción>
Datos específicos	<Datos Específicos>
Estabilidad	<Estabilidad>
Comentarios	<Comentarios>

Tabla 6. Patrón de Requisito

- **Identificador:** incluye el tipo de requisito y el número identificador consecutivo, comenzaremos por el número 1.
 - **RF-N:** para los requisitos funcionales utilizamos las siglas RF acompañado de un número N consecutivo.
 - **RNF-N:** se usan las siglas RNF para los requisitos no funcionales, donde la N es el número consecutivo correspondiente.
- **Título:** nombre descriptivo de la funcionalidad del requisito.
- **Prioridad:** se establece una prioridad atendiendo a la importancia del requisito dentro del sistema. Las prioridades pueden ser: Alta, Media y Baja.
- **Entrada:** Son los datos que el sistema necesita para iniciar la operación correspondiente.
- **Salida:** Datos que se obtienen como salida una vez procesada la información de entrada.
- **Descripción:** Se detalla las acciones a realizar por parte del usuario para llevar a cabo el requisito establecido.
- **Datos específicos:** Se indican en detalle cada uno de los datos necesarios para ejecutar la operación determinada.
- **Estabilidad:** Indica el nivel de estabilidad que debe tener en el sistema para que no se produzcan errores o estos sean los mínimos posibles. Los niveles pueden ser Alto, Medio y Bajo
- **Comentarios:** Información adicional necesaria a tener en cuenta para iniciar o finalizar con éxito la acción descrita.

4.2.1 Requisitos Funcionales

Detallamos los requisitos funcionales de la aplicación:

RF - 001	
Título	Gestión de Asignaturas (Alta)
Prioridad	Alta
Entrada	Datos de la Asignatura
Salida	Alta de la asignatura en la base de datos
Descripción	Para la gestión de altas de asignaturas, el administrador rellenará un formulario de Alta con los datos solicitados.
Datos específicos	Nombre de la Asignatura
Estabilidad	Alta
Comentarios	Al dar de alta la asignatura se comprobará que la asignatura no exista en la base de datos.

Tabla 7. Requisito Funcional - Alta de Asignaturas

RF - 002	
Título	Gestión de Asignaturas (Baja)
Prioridad	Alta
Entrada	Datos de la Asignatura
Salida	Eliminación de la asignatura en la base de datos
Descripción	La acción a realizar para dar de baja una asignatura, es pinchando en el botón Eliminar que está disponible al consultar los datos de la asignatura a suprimir de la base de datos.
Datos Específicos	Nombre de la Asignatura
Estabilidad	Alta
Comentarios	Al dar de baja una asignatura se comprobará previamente que se haya dado de baja las subasignaturas y videos asociados a la asignatura que se pretende dar de baja.

Tabla 8. Requisito Funcional - Baja de Asignaturas

RF - 003	
Título	Gestión de Asignaturas (Modificación)
Prioridad	Alta
Entrada	Datos de la asignatura
Salida	Se modificará los datos que el administrador haya solicitado
Descripción	Se puede modificar el nombre de la asignatura, de las subasignaturas asociadas a la asignatura, además de los videos asociados a la misma, tanto para las subasignaturas como para los videos tendremos disponible una lista desplegable en la que aparecerán el nombre de la asignatura o el título del video, estos campos solo pueden ser modificados por las subasignaturas o videos que hayan sido dados de alta previamente.
Datos Específicos	Nombre de la Asignatura, Nombre de Subasignatura y Título de Videos
Estabilidad	Alta
Comentarios	Se modificará los datos de la Asignatura, previa comprobación de que la asignatura existe y que las modificaciones son permitidas por la base de datos.

Tabla 9. Requisito Funcional - Modificación de Asignaturas

RF - 004	
Título	Gestión de Asignaturas (Listado)
Prioridad	Alta
Entrada	Petición de Listado
Salida	Listado de Asignaturas
Descripción	Al pinchar en el botón Gestionar Asignatura nos proporciona un listado de las mismas.
Estabilidad	Alta
Comentarios	Por defecto el listado se ordenará alfabéticamente por nombre de Asignatura

Tabla 10. Requisito Funcional - Listado de Asignaturas

RF - 005	
Título	Gestión de Asignaturas (Consulta)
Prioridad	Alta
Entrada	Petición de Consulta
Salida	Datos de la asignatura
Descripción	Una vez se observe el listado de las asignaturas podemos pinchar sobre la que deseamos consultar y a continuación visualizaremos sus datos.
Datos Específicos	Nombre de la asignatura, nombres de las subasignaturas asociadas y títulos de los videos asociados.
Estabilidad	Alta
Comentarios	Una vez seleccionemos la asignatura a consultar se observará en una nueva vista el nombre de la asignatura, nombres de la subasignaturas asociadas a la asignatura y los títulos de los videos asociados a ella.

Tabla 11. Requisito Funcional - Consulta de Asignaturas

RF - 006	
Título	Gestión de SubAsignaturas (Alta)
Prioridad	Alta
Entrada	Datos de la SubAsignatura
Salida	Alta de la subAsignatura en la base de datos
Descripción	Para la gestión de altas de subasignaturas, el administrador rellenará un formulario de Alta con los datos solicitados.
Datos específicos	Nombre de la SubAsignatura, Nombre de la Asignatura
Estabilidad	Alta
Comentarios	Al dar de alta la subAsignatura se comprobará que no exista en la base de datos y además es necesario que sea asociada a una asignatura.

Tabla 12. Requisito Funcional - Alta de SubAsignaturas

RF - 007	
Título	Gestión de SubAsignaturas (Baja)
Prioridad	Alta
Entrada	Datos de la SubAsignatura
Salida	Eliminación de la subasignatura en la base de datos
Descripción	La acción a realizar para dar de baja una subasignatura, es pinchando en el botón Eliminar que está disponible al consultar los datos de la subasignatura a suprimir de la base de datos.
Datos Específicos	Nombre de Subasignatura
Estabilidad	Alta
Comentarios	Al dar de baja una Subasignatura se comprobará previamente que se haya dado de baja los videos asociados a ella.

Tabla 13. Requisito Funcional - Baja de SubAsignaturas

RF - 008	
Título	Gestión de SubAsignaturas (Modificación)
Prioridad	Alta
Entrada	Datos de la subasignatura
Salida	Se modificará los datos que el administrador ha solicitado
Descripción	Al modificar los datos de la SubAsignatura podemos cambiar su nombre y el nombre de la asignatura a la que pertenece, este nombre aparecerá en una lista desplegable de la cuál elegiremos la opción por la que se desea sustituir. Se presenta en una lista desplegable ya que solo es posible cambiar el nombre de la asignatura por las que hayan sido previamente dadas de alta en el sistema.
Datos Específicos	Nombre de la Asignatura y SubAsignatura
Estabilidad	Alta
Comentarios	Se modificará los datos de la SubAsignatura, previa comprobación de que la subasignatura existe y que las modificaciones son permitidas por la base de datos

Tabla 14. Requisito Funcional - Modificación de SubAsignaturas

RF- 009	
Título	Gestión de SubAsignaturas (Listado)
Prioridad	Alta
Entrada	Al pinchar en el botón Gestionar SubAsignaturas nos proporciona un listado de todas las subasignaturas registradas, además en el mismo listado se puede observar a que asignatura pertenece cada asignatura.
Salida	Listado de SubAsignaturas
Descripción	Proporciona una listado de las subasignaturas almacenadas en el sistema.
Estabilidad	Alta
Comentarios	Por defecto el listado se ordenará alfabéticamente por nombre de SubAsignatura.

Tabla 15. Requisito Funcional - Listado de SubAsignaturas

RF - 010	
Título	Gestión de SubAsignaturas (Consulta)
Prioridad	Alta
Entrada	Petición de Consulta
Salida	Datos de la SubAsignatura
Descripción	Una vez veamos el listado de todas las subasignaturas registradas es posible seleccionar la que deseamos consultar, cuando se haya elegido se mostrarán todos los datos en una nueva vista.
Datos Específicos	Nombre de la SubAsignatura y Nombre de Asignatura
Estabilidad	Alta
Comentarios	Al seleccionar la subasignatura a consultar observamos su nombre y la asignatura a la que está asociada.

Tabla 16. Requisito Funcional - Consulta de SubAsignaturas

RF - 011	
Título	Gestión de Videos (Alta)
Prioridad	Alta
Entrada	Datos de los Videos
Salida	Alta del video en la base de datos.
Descripción	Para la gestión de alta de Video, el administrador rellenara un formulario de Alta con los datos solicitados.
Datos específicos	Título, descripción, asignatura, subasignatura, urlVideo y preguntas.
Estabilidad	Alta
Comentarios	Se procede a dar de alta el video indicado, para que no haya duplicidad de títulos, se ha preferido indicar que el título sea único, esto permite que el video se identifique de manera unívoca mediante el título.

Tabla 17. Requisito Funcional - Alta de Videos

RF - 012	
Título	Gestión de Videos (Baja)
Prioridad	Alta
Entrada	Datos de los Videos
Salida	Eliminación de los datos del video en la base de datos.
Descripción	La acción a realizar para dar de baja un video, es pinchando en el botón Eliminar que está disponible al consultar los datos del video a suprimir de la base de datos
Datos específicos	Título, descripción, urlVideo.
Estabilidad	Alta
Comentarios	Para proceder a dar de baja un video es necesario previamente haber dado de baja todas las preguntas que tenga asociadas.

Tabla 18. Requisito Funcional - Baja de Videos

RF - 013	
Título	Gestión de Videos (Modificación)
Prioridad	Alta
Entrada	Datos de los Videos
Salida	Se modificarán los datos que el administrador ha solicitado
Descripción	Se pueden modificar todos los datos que identifican al video, es necesario indicar que para rectificar tanto para el nombre de la asignatura como el nombre de la subasignatura aparecerá un desplegable con las opciones a cambiar.
Datos específicos	Título, descripción, asignatura, subasignatura, urlVideo, preguntas
Estabilidad	Alta
Comentarios	Al modificar una video se puede modificar todos los campos, también es posible rectificar preguntas asociadas al video

Tabla 19. Requisito Funcional - Modificación de Videos

RF - 014	
Título	Gestión de Videos (Listado)
Prioridad	Alta
Entrada	Al pinchar en el botón Gestionar Videos nos proporciona un listado de todos los videos registrados en el sistema.
Salida	Listado de Videos
Descripción	Proporciona una listado de los videos almacenadas en el sistema
Estabilidad	Alta
Comentarios	Por defecto el listado se ordenará alfabéticamente por Titulo de Video.

Tabla 20. Requisito Funcional - Listado de Videos

RF - 015	
Título	Gestión de Videos (Consulta)
Prioridad	Alta
Entrada	Petición de Consulta
Salida	Datos de los Videos
Descripción	Una vez vemos el listado de todos los Videos registrados, es posible seleccionar el video a consultar, cuando se haya elegido se mostrarán todos los datos en una nueva vista.
Datos Específicos	Título, Descripción, Asignatura, SubAsignatura, UrlVideo, Preguntas
Estabilidad	Alta
Comentarios	Al seleccionar el video a consultar observaremos todos sus datos además de las preguntas asociadas a él.

Tabla 21. Requisito Funcional - Consulta de Videos

RF - 016	
Título	Gestión de Preguntas (Alta)
Prioridad	Alta
Entrada	Datos de las Preguntas
Salida	Alta de la Pregunta en la base de datos
Descripción	Para la gestión de alta de Preguntas, el administrador rellenara un formulario de Alta con los datos solicitados.
Datos específicos	Enunciado, OrdenPregunta, Correcto, Respuesta1, Respuesta2, Respuesta3, Respuesta4, Video
Estabilidad	Alta
Comentarios	Se procede a dar de alta a la pregunta siempre que esté asociada a un video de los que tenemos registrados, es importante indicar que el orden de pregunta es el tiempo (minutos) de la reproducción del video en el que debe aparecer la pregunta de forma automática, momento en el cual se interrumpirá la reproducción normal del video.

Tabla 22. Requisito Funcional - Alta de Preguntas

RF - 017	
Título	Gestión de Preguntas (Baja)
Prioridad	Alta
Entrada	Datos de las Preguntas.
Salida	Eliminación de los datos de las preguntas registrados en la base de datos.
Descripción	La acción a realizar para dar de baja una pregunta, es pinchando en el botón Eliminar que está disponible al consultar los datos de la pregunta a suprimir de la base de datos.
Datos específicos	Enunciado, OrdenPregunta, Correcto, Respuesta1, Respuesta2, Respuesta3, Respuesta4.
Estabilidad	Alta
Comentarios	Se procederá a dar de baja a la pregunta solicitada.

Tabla 23. Requisito Funcional - Baja de Preguntas

RF - 018	
Título	Gestión de Preguntas(Modificación)
Prioridad	Alta
Entrada	Datos de Preguntas
Salida	Se modificarán los datos de las preguntas que el administrador ha solicitado.
Descripción	Se puede modificar todos los datos de las preguntas, incluido el video al que están asociados.
Datos específicos	Enunciado, OrdenPregunta, Correcto, Respuesta1, Respuesta2, Respuesta3, Respuesta4,Video
Estabilidad	Alta
Comentarios	Al modificar las preguntas es muy importante tener mucho cuidado cuando deseemos cambiar el campo Orden Pregunta, ya que es un campo importante para el correcto funcionamiento de la aplicación.

Tabla 24. Requisito Funcional - Modificación de Preguntas

RF - 019	
Título	Gestión de Preguntas (Listado)
Prioridad	Alta
Entrada	Al pinchar en el botón Gestionar Preguntas nos proporciona un listado de todas las preguntas registradas, además es posible observar los videos a las que están asociadas.
Salida	Listado de Preguntas
Descripción	Proporciona una listado de las preguntas almacenadas en el sistema
Estabilidad	Alta
Comentarios	Por defecto el listado se ordenará alfabéticamente por el campo Enunciado de Pregunta.

Tabla 25. Requisito Funcional - Listado de Preguntas

RF- 020	
Título	Gestión de Preguntas (Consulta)
Prioridad	Alta
Entrada	Petición de Consulta
Salida	Datos de las Preguntas
Descripción	Una vez vemos el listado de las Preguntas registradas podemos seleccionar el enlace del enunciado de la pregunta a consultar, cuando se haya escogido se mostrarán todos los datos en una nueva vista.
Datos Específicos	Enunciado, OrdenPregunta, Correcto, Respuesta1, Respuesta2, Respuesta3, Respuesta4,Video
Estabilidad	Alta
Comentarios	Al consultar la pregunta seleccionada, podemos observar todos sus datos además del video al que encuentra asociada.

Tabla 26. Requisito Funcional - Consulta de Preguntas

RF - 021	
Título	Gestión de Búsqueda General
Prioridad	Alta
Entrada	Palabra Clave a buscar
Salida	Listado de Videos
Descripción	Al introducir la palabra clave, el sistema devuelve todos los videos que contengan la palabra buscada en los campos: título, descripción, asignatura y subasignatura.
Datos Específicos	Término de Búsqueda
Estabilidad	Alta
Comentarios	Al consultar por la palabra clave podremos obtener un listado de los videos que contengan el tema buscado y reproducir el que este acorde a nuestras necesidades.

Tabla 27. Requisito Funcional - Búsqueda General

RF - 022	
Título	Gestión de Búsqueda por SubAsignaturas
Prioridad	Alta
Entrada	Elección de la SubAsignatura disponible en el catálogo clasificado por Asignaturas
Salida	Listado de Videos clasificados por SubAsignaturas
Descripción	Al clasificarse los videos por las subasignaturas que pertenecen a una asignatura, es posible obtener todos los videos que han sido organizados por una subasignatura determinada. Para obtener tales resultados, es necesario haber seleccionado previamente la subasignatura del catálogo.
Datos Específicos	Nombre de SubAsignatura
Estabilidad	Alta
Comentarios	En el catálogo de subasignaturas, agrupadas a la vez por las asignaturas, existe un link creado con el nombre de la subasignatura, este link debe ser pulsado para obtener el listado de todos los videos clasificados por la subasignatura a consultar.

Tabla 28. Requisito Funcional - Búsqueda de SubAsignaturas

RF - 023	
Título	Gestión de Interacción con los Videos
Prioridad	Alta
Entrada	Elección del Video
Salida	Reproducción del Video y visualización de las preguntas.
Descripción	Al reproducir el video seleccionado irán apareciendo las preguntas automáticamente en los tiempos establecidos de tal forma que interrumpirán la reproducción normal del video. Estas preguntas deben ser contestadas por el propio usuario, he aquí la interacción que ofrece la aplicación en el momento mismo de la reproducción del video.
Datos Específicos	Título del Video
Estabilidad	Alta
Comentarios	La reproducción secuencial del video está condicionada por el tipo de respuesta que se den a las preguntas.

Tabla 29. Requisito Funcional - Interacción con Videos

RF - 024	
Título	Formulario de Colaboración
Prioridad	Alta
Entrada	Datos del Formulario de Colaboración
Salida	Email enviado a la aplicación
Descripción	Se da la oportunidad a las personas que desean cedernos sus videos, con el propósito académico, de rellenar un formulario de colaboración para ponernos en contacto con ellas.
Datos Específicos	Nombre, Apellidos, Email, Profesión, Sexo, Comentario
Estabilidad	Alta
Comentarios	Los datos del Formulario de Colaboración serán enviados vía email al administrador de la aplicación quién mantendrá contacto por la misma vía, una vez hayamos establecido comunicación con el usuario colaborador le indicaremos la forma de hacernos llegar los videos, además de informarle los requisitos que deben cumplir las preguntas. En el momento que dispongamos del video, nos aseguramos que los videos tienen el único propósito de enseñar.

Tabla 30. Requisito Funcional - Formulario de Colaboración

4.2.2 Requisitos No Funcionales

Detallamos los requisitos de tipo no funcional necesarios:

RNF - 001	
Título	Funcionamiento con Internet
Prioridad	Alta
Descripción	Es necesaria una conexión a internet para el uso de la aplicación
Estabilidad	Alta

Tabla 31. Requisito No Funcional - Funcionamiento con Internet

RNF - 002	
Título	Funcionamiento con Navegador Web
Prioridad	Alta
Descripción	Es necesario un navegador Web para acceder a la aplicación
Estabilidad	Alta

Tabla 32. Requisito No Funcional - Funcionamiento con Navegador Web

RNF - 003	
Título	Compatibilidad con HTML5
Prioridad	Alta
Descripción	Es necesario un navegador Web compatible con HTML5
Estabilidad	Alta

Tabla 33. Requisito No Funcional - Compatibilidad con HTML5

RNF - 004	
Título	Compatibilidad con Mozilla Firefox
Prioridad	Alta
Descripción	La aplicación será compatible con Mozilla Firefox versión 9 o superior
Estabilidad	Alta

Tabla 34. Requisito No Funcional - Compatibilidad con Mozilla Firefox

RNF - 005	
Título	Compatibilidad con Google Chrome
Prioridad	Alta
Descripción	La aplicación será compatible con Chrome versión 16 o superior
Estabilidad	Alta

Tabla 35. Requisito No Funcional - Compatibilidad con Google Chrome

RNF - 006	
Título	Compatibilidad con Internet Explorer
Prioridad	Alta
Descripción	La aplicación será compatible con Internet Explorer versión 9 o superior
Estabilidad	Alta

Tabla 36. Requisito No Funcional - Compatibilidad con Internet Explorer

RNF - 007	
Título	Compatibilidad con Safari
Prioridad	Alta
Descripción	La aplicación será compatible con Safari versión 5.1 o superior
Estabilidad	Alta

Tabla 37. Requisito No Funcional - Compatibilidad con Safari

RNF - 008	
Título	Framework
Prioridad	Alta
Descripción	La aplicación será desarrollado con Framework Grails
Estabilidad	Alta

Tabla 38. Requisito No Funcional - Framework

RNF - 009	
Título	Lenguaje de Programación
Prioridad	Alta
Descripción	Se desarrolla con Groovy como lenguaje de programación
Estabilidad	Alta

Tabla 39. Requisito No Funcional - Lenguaje de Programación

RNF - 010	
Título	Entorno de Desarrollo Integrado (IDE)
Prioridad	Alta
Descripción	SpringSource Tool Suite como IDE de la aplicación
Estabilidad	Alta

Tabla 40. Requisito No Funcional - Entorno de Desarrollo

RNF - 011	
Título	Base de Datos
Prioridad	Alta
Descripción	MySQL Workbench como base de datos de la aplicación
Estabilidad	Alta

Tabla 41. Requisito No Funcional - Base de Datos

RNF - 012	
Título	Servidor
Prioridad	Alta
Descripción	La aplicación utilizará Apache Tomcat integrado en el IDE
Estabilidad	Alta

Tabla 42. Requisito No Funcional - Servidor

RNF - 013	
Título	Formato Videos válidos para HTML5
Prioridad	Alta
Descripción	Webm, OGG y MP4 formatos de Video compatibles con HTML5
Estabilidad	Alta

Tabla 43. Requisito No Funcional - Formato de Videos para HTML5

4.3 Especificación de casos de uso

Especificar un caso de uso de la aplicación es determinar un empleo de la aplicación llevada a cabo por un tipo determinado de usuario, se ha establecido la metodología UML (Lenguaje de Modelo Unificado) que permiten graficar los componentes del sistema y la estructura de su funcionamiento de una forma clara y concisa, de esta manera permite ver las diferentes acciones y relaciones entre ellas, además de los actores que participan en cada una.

Los componentes de la representación gráfica de cada caso de uso son los siguientes:

- **Actor:** representa el tipo de usuario que realiza las acciones en el sistema.
- **Caso de Uso:** tarea realizada por el actor o por otro caso de uso dentro del sistema.
- **Relaciones:** nexo entre los actores y los casos de uso, también se puede dar la conexión entre dos casos de uso, las relaciones pueden ser:
 - **Asociación:** representa la posibilidad de que un usuario realice un caso de uso.
 - **Dependencia:** representa la relación entre dos casos de uso.
 - **Generalización:** representa el uso o la herencia entre casos de uso.
- **Escenario:** nombre del contexto de los casos de uso.

Para la representación de los diagramas se va a seguir el siguiente esquema:



Figura 18. Nomenclatura de Diagrama de Uso

Para la descripción de los casos de uso se acompañarán de las siguientes tablas descriptivas que recogen la siguiente información.

Identificador	<Identificador>
Nombre	<nombre>
Actor	<actor>
Objetivos	<objetivo>
Precondiciones	<precondiciones>
Postcondiciones	<postcondiciones>
Escenario básico	<escenario>

Figura 19. Patrón de descripción de Caso de Uso

- **Identificador:** identifica el caso de uso, estará compuesto por una letra que identifique al actor que realiza la acción, y un número único secuencial para cada tipo de actor. Citamos un ejemplo A-001, la letra A identifica al actor y 001 es el número único para ese actor.
 - Administrador: se identifica con la letra A
 - Usuario Consumidor: se identifica con UC
 - Usuario Colaborador: identificamos con UCO
- **Nombre:** detalla el nombre característico que identifica al caso de uso.
- **Actor:** se indica el actor que realiza la acción.
- **Objetivo:** propósito de la acción.
- **Precondición:** condiciones previas para poder realizar la acción.
- **Postcondiciones:** condiciones posteriores para que la acción se realice correctamente.
- **Escenario básico:** pasos a seguir para llevar a cabo la acción.

A continuación se detallan los diagramas de los casos de uso que corresponden a los 3 actores que interactúan con la aplicación: administrador, usuario consumidor y usuario colaborador

4.3.1 Administrador del Sistema

Diagrama de caso de uso del Administrador:

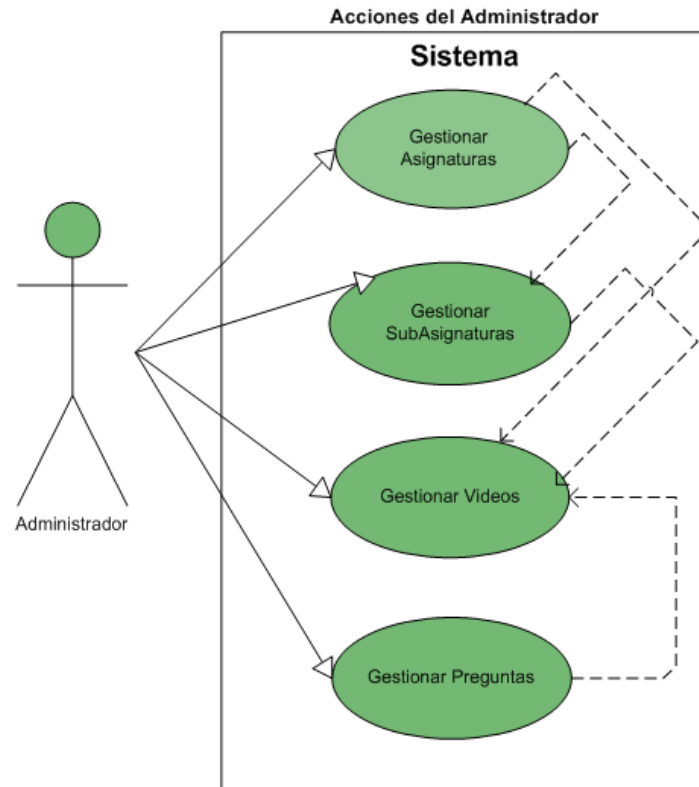


Figura 20. Diagrama de Caso de Uso del Administrador del Sistema

- **Gestión de Asignaturas**

Identificador	A-001
Nombre	Gestión de Asignaturas
Actor	Administrador
Objetivos	Permite al administrador crear, borrar, modificar, consultar y listar las asignaturas registradas en la aplicación
Precondiciones	Acceder a la aplicación y autenticarse con una cuenta de administrador
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. El administrador se identifica mediante su login y su contraseña. 2. Accede a la sección de gestión de asignaturas. 3. Realiza las operaciones necesarias.

Tabla 44. Descripción del Caso de Uso - Gestión de Asignaturas

- **Gestión de SubAsignaturas**

Identificador	A-002
Nombre	Gestión de SubAsignaturas
Actor	Administrador
Objetivos	Permite al administrador crear, borrar, modificar, consultar y listar las subasignaturas registradas en la aplicación.
Precondiciones	Acceder a la aplicación y autenticarse con una cuenta de administrador
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. El administrador se identifica mediante su login y su contraseña. 2. Accede a la sección de gestión de subasignaturas. 3. Realiza las operaciones necesarias.

Tabla 45. Descripción del Caso de Uso - Gestión de SubAsignaturas

- **Gestión de Videos**

Identificador	A-003
Nombre	Gestión de Videos
Actor	Administrador
Objetivos	Permite al administrador crear, borrar, modificar, consultar y listar los videos que forman parte de la aplicación
Precondiciones	Acceder a la aplicación y autenticarse con una cuenta de administrador
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. El administrador se identifica mediante su login y su contraseña. 2. Accede a la sección de gestión de subasignaturas. 3 .Realiza las operaciones necesarias.

Tabla 46. Descripción del Caso de Uso - Gestión de Videos

- **Gestión de Preguntas**

Identificador	A-004
Nombre	Gestión de Preguntas
Actor	Administrador
Objetivos	Permite al administrador crear, borrar, modificar, consultar y listar las preguntas
Precondiciones	Acceder a la aplicación y autenticarse con una cuenta de administrador
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. El administrador se identifica mediante su login y su contraseña. 2. Accede a la sección de gestión de subasignaturas. 3. Realiza las operaciones necesarias.

Tabla 47. Descripción del Caso de Uso - Gestión de Preguntas

4.3.2 Usuarios Consumidor

Diagrama del caso de uso del Usuario Consumidor:

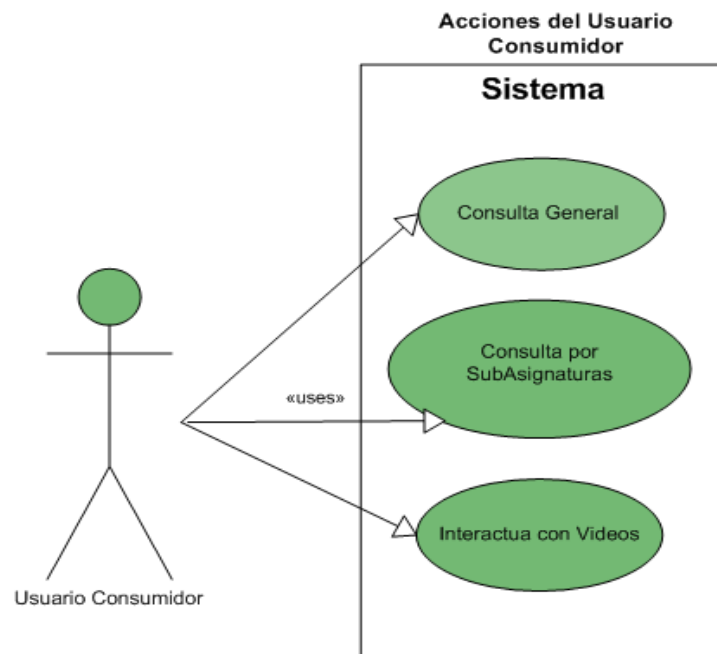


Tabla 48. Diagrama de Caso de Uso del Usuario Consumidor del Sistema

- **Consulta General**

Identificador	UC-001
Nombre	Consulta General
Actor	Usuario Consumidor
Objetivos	El usuario hace una búsqueda con una palabra clave en el sistema, este devolverá una lista con los videos que contengan el término buscado
Precondiciones	Introducir una término de búsqueda
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. Se escribe el término de búsqueda en el recuadro correspondiente, a continuación pulsamos el botón buscar 2. El sistema proporciona los videos que contengan el término de búsqueda. 3. Se selecciona el video que nos interese reproducir. 4. Para reproducir el video seleccionado pulsamos el botón Play.

Tabla 49. Descripción del Caso de Uso - Consulta General

- **Consulta Por SubAsignaturas**

Identificador	UC-002
Nombre	Consulta por SubAsignaturas
Actor	Usuario Consumidor
Objetivos	El sistema presenta un menú donde están clasificadas las asignaturas y estas a la vez están compuestas por subasignaturas que pertenecen a cada una de ellas, de esta manera se indica al usuario consumidor los videos de las Asignaturas y Subasignaturas que tiene disponible el sistema.
Precondiciones	Pulsar el enlace de la SubAsignatura.
Postcondiciones	N/A
Escenario básico	<ol style="list-style-type: none"> 1. En el catálogo o menú se pulsa la Asignatura a consultar, en ese momento aparece un desplegable con la lista de las Subasignaturas que forman parte de ella.

	<p>2. Pulsamos la subasignatura a consultar.</p> <p>3. El sistema proporciona los videos que han sido categorizados por esa subasignatura.</p> <p>4. Se selecciona el video que nos interese visualizar.</p> <p>5. Para reproducir el video seleccionado pulsamos el botón Play.</p>
--	--

Tabla 50. Descripción del Caso de Uso - Consulta por SubAsignaturas

- **Interactúa con Videos**

Identificador	UC-003
Nombre	Interacción con los Videos
Actor	Usuario Consumidor
Objetivos	Al pinchar sobre el botón Play del video seleccionado este comenzará su reproducción, acto seguido se producirán interrupciones en la reproducción normal del video por la presentación de las preguntas que aparecerán de forma automática.
Precondiciones	Seleccionar el video a reproducir.
Postcondiciones	Contestar a todas las preguntas.
Escenario básico	<p>1. Pulsamos el botón Play para iniciar la reproducción.</p> <p>2. La reproducción se producirá de forma normal hasta que aparezca una pregunta en el tiempo establecido.</p> <p>3. La pregunta aparecerá en una ventana emergente con el enunciado y las posibles respuestas.</p> <p>4. El usuario deberá seleccionar una respuesta y pinchar en el botón "Comprobar Respuesta".</p> <p>5. El sistema indicará si la respuesta es correcta e incorrecta, mediante un mensaje.</p> <p>6. Una vez hayamos conocido si la respuesta es correcta o incorrecta pinchamos el botón "Acepta", con esto la ventana se cierra.</p> <p>7. Al volver al video después de que se haya cerrado la ventana emergente en la que aparece la pregunta, debemos pulsar el botón Play, si la respuesta ha sido contestada correctamente el</p>

	<p>video continuará con la reproducción normal, en caso contrario se volverá a reproducir el tramo del video segundo después del que se haya contestado la última pregunta de forma correcta.</p> <p>8. Una vez contestadas todas las preguntas de forma correcta se terminará con la reproducción dando la oportunidad de repetir el video.</p>
--	--

Tabla 51. Descripción del Caso de Uso - Interactúa con Videos

4.3.3 Usuarios Colaborador

Diagrama de caso del uso del Usuario Colaborador:

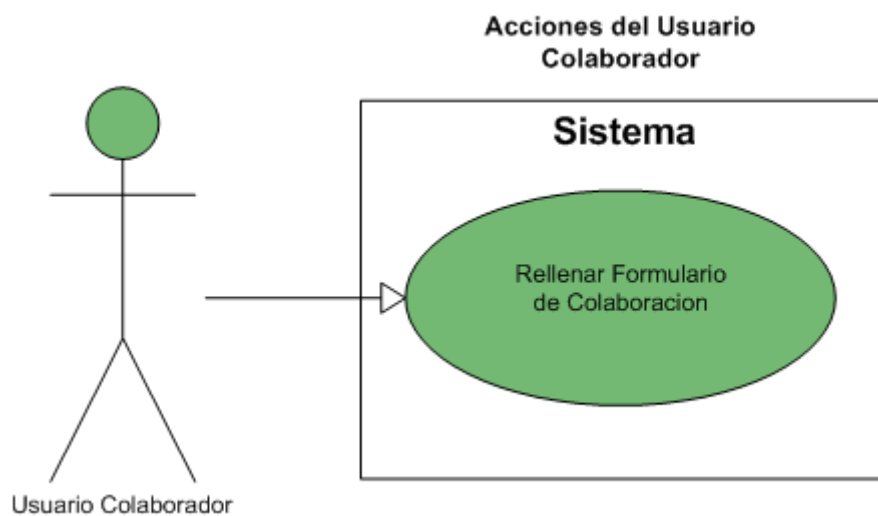


Tabla 52. Diagrama de Caso de Uso del Usuario Colaborador con el Sistema

- **Rellenar Formulario de Colaboración**

Identificador	UCO-001
Nombre	Rellenar Formulario de Colaboración
Actor	Usuario Colaborador
Objetivos	Mediante este Formulario de Colaboración permitimos a las personas que desean cedernos sus videos con contenido académico se pongan en contacto con nosotros para indicarles la forma de hacernos llegar los videos y sus preguntas
Precondiciones	Acceder al formulario de la colaboración.
Postcondiciones	Introducir datos correctos del usuario en el formulario de colaboración.
Escenario básico	1. Accedemos al Formulario de colaboración pulsando el botón

	<p>" Participar" disponible en la página principal.</p> <p>2. Rellenar todos los datos solicitados y pulsar el botón enviar.</p> <p>3. El sistema enviará automáticamente un email al administrador de AprenVid con los datos del formulario.</p> <p>4. El administrador mantendrá contacto con el Usuario vía email, en ese momento el administrador le indicará al usuario colaborador las condiciones que deben cumplir tanto los video y las preguntas, además de comentarle como nos puede hacer esta información.</p>
--	---

Tabla 53. Descripción del Caso de Uso - Rellenar Formulario de Colaboración

CAPÍTULO 5

5. Diseño de la Aplicación

En este capítulo se detalla el diseño de la aplicación, contempla tanto la arquitectura de la aplicación, el diseño del modelo de datos, y la interfaz gráfica que visualiza el usuario.

5.1 Arquitectura de la aplicación

Se incluyen dos tipos de arquitecturas bien diferenciadas. Por un lado tenemos la arquitectura Cliente Servidor, esta es la que ven los usuarios, típica de las aplicaciones web, y por otro lado la arquitectura interna de la aplicación basada en el patrón Vista Controlador, la que ve el desarrollador.

5.1.1 Arquitectura Cliente-Servidor

Arquitectura que contiene lo que su nombre indica:

- **Servidor:** El Servidor que contiene la aplicación o el servicio a ofrecer, se encuentra a la espera de recibir peticiones y servirlos.
- **Cliente:** Es el encargado de realizar peticiones a través de una red de comunicación al servidor.

En las aplicaciones Web, el cliente a través de los navegadores que tenemos disponibles en la actualidad y de una conexión a internet realiza una petición introduciendo la URL de la aplicación. La petición llega al servidor y este sirve el recurso solicitado.

En nuestra aplicación las peticiones vendrán dadas solo por los usuarios.

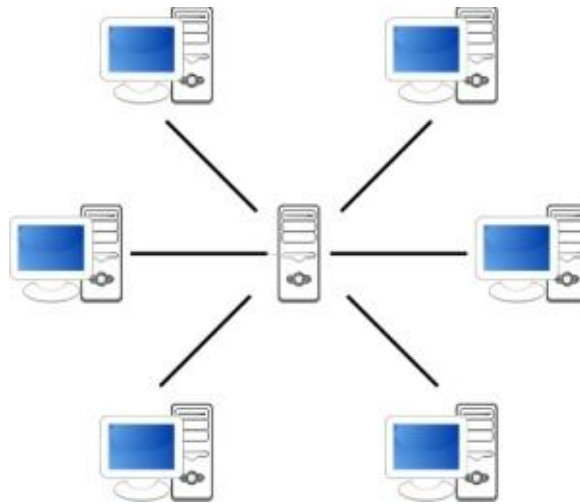


Figura 21. Arquitectura Cliente Servidor

5.1.2 Modelo Vista Controlador

Arquitectura Software que estructura un determinado sistema en tres capa:

- **Modelo de Datos:** Representación de los diferentes datos almacenados en la base de datos, y los mecanismos de acceso a esta información.
- **Vista:** representa el conjunto de interfaces que ven los usuarios a través del navegador.
- **Controladores:** mecanismos encargados de interactuar entre la vista y el modelo de datos.

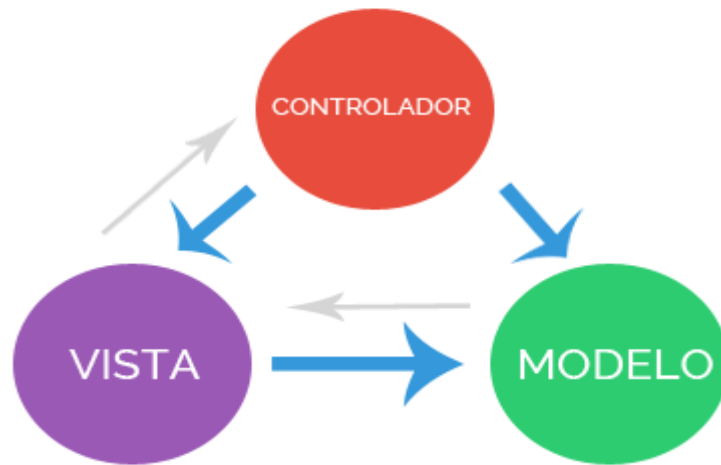


Figura 22. Modelo Vista Controlador

Cuando un usuario realiza cualquier acción sobre la vista, son los controladores los encargados de realizar la opción requerida, y si corresponde realiza consultas o modificaciones sobre los datos, aunque también puede ocurrir que el modelo interactúe directamente con las vistas sin pasar directamente por el controlador.

A día de hoy existen muchos Frameworks, a continuación mostramos una tabla con algunos ejemplos que siguen el modelo Vista-Controlador.

Lenguaje	Licencia	Nombre
Objective C	Apple	Cocoa
Ruby	MIT	Ruby on Rails
Java	Apache	Grails
Java	GPL	Interface Java Objects
Java	Apache	Struts
Java	Apache	Spring
JavaScript	GPLv3	Sails.JS
Perl	GPL	Mojolicious
PHP	GPL	BitPHP
PHP	[1]	Yii
PHP	MIT	Laravel
Python	ZPL	Zope3
Python	GPL	Web2py
.NET	Castle Project	MonoRail
.NET	Apache	Spring.NET

Tabla 54. Framework con Arquitectura MVC

Tal y como se puede ver en la tabla, Grails es uno de los Framework que siguen ese tipo de Modelo.

- **Ventajas del Modelo Vista Controlador**

A continuación se citan las ventajas que ofrecen los framework de los ejemplo citados en la tabla superior.

1. Podrás dividir la lógica de negocio del diseño, haciendo tu proyecto más escalable.
2. Te facilitará el uso de URL amigables, importantes para el SEO (Posicionamiento web), la mayoría de frameworks MVC lo controlan.
3. Muchos frameworks MVC ya incluyen librerías de Javascript como JQuery, lo que te facilitará validar formularios (Ej. JQuery.Validate) en el cliente y en el servidor.
4. Puedes utilizar abstracción de datos, como lo hace Ruby on Rails o con frameworks como Hibernate para Java o NHibernate para ASP .NET MVC, facilitando la realización de consultas a la base de datos.
5. La mayoría de frameworks controlan el uso de la memoria Caché, hoy en día muy importante para el posicionamiento web, ya que buscadores como google dan prioridad a las webs que tengan menor tiempo de descarga.
6. En el caso de proyectos donde hay varios desarrolladores, el seguir métodos comunes de programación, hace que el código sea más entendible entre ellos, pudiendo uno continuar el trabajo del otro. En estos casos es conveniente utilizar herramientas de control de versiones como Subversión.
7. Los frameworks están creados para facilitar el trabajo de los desarrolladores, encontrarás clases para controlar fechas, URL's, Webservices, etc. lo que tiene una gran ventaja en cuanto a productividad. Inicialmente como es lógico habrá una curva de aprendizaje, pero luego tendrás muchos beneficios.
8. Poco a poco el desarrollo web se orienta a lo que se denomina "Agile Web Development" (Desarrollo ágil de aplicaciones web), con frameworks como Ruby on Rails que ayudan a crear proyectos de calidad y en corto tiempo. Existen varios frameworks en PHP e incluso ASP .NET que en su nueva versión ya contempla el MVC con Visual C#.
9. Utilizar herramientas con tecnología escalable hace más atractivo tu proyecto en caso de buscar inversión externa, muchas veces para hacer crecer un proyecto, es necesario buscar socios o Bussines Angels que te ayuden a impulsarlo.
10. Un Framework MVC te ayuda a controlar los recursos del servidor, evitando Bugs que puedan repercutir en el rendimiento, por ejemplo, muchas veces olvidamos cerrar conexiones a la base de datos, sobrecargando el servidor.

5.2 Modelo de datos

El modelo de datos representa las diferentes tablas que almacenan los datos necesarios para el funcionamiento de la aplicación, así como las relaciones existentes entre las distintas tablas.

5.2.1 UML

UML son las siglas de Unified Modeling Language (Lenguaje unificado de modelado), es un lenguaje visual que permite ver, documentar, especificar y construir sistemas. Este lenguaje esta soportado por Object Management Group que es un consorcio dedicado al cuidado de varios estándares orientados a objetos, como por ejemplo, XMI y CORBA.

Con UML definiremos como se relacionan las entidades de dominio de la aplicación, y que atributos tienen estas entidades. Para ello, se utilizarán los siguientes elementos:

- **Clase de dominio:** describe el objeto que representa.
- **Atributos:** son las distintas propiedades con las que cuenta un objeto. Por ejemplo, un objeto coche, tiene atributos como peso, altura, color, etc.
- **Relación:** define como se relacionan diferentes clases de dominio. Las relaciones son bidireccionales pero no se leen de la misma forma en un sentido que en otro.
- **Numeración de la relación:** especifica cuantos objetos de una de las partes de la relación pueden interactuar con el elemento del otro extremo de la relación.

En la siguiente imagen podemos ver un ejemplo de cómo un objeto "coche", tiene varios atributos, como son peso y color (entre otros muchos que puede tener) y a su vez tiene objetos rueda. Un objeto rueda tiene atributos como el radio, la anchura o la presión. En la relación entre coche y rueda, podemos ver que un coche puede tener de 0 a n ruedas. Si leyéramos la relación al revés, esta se leería así: una rueda es "tenida" por un coche.

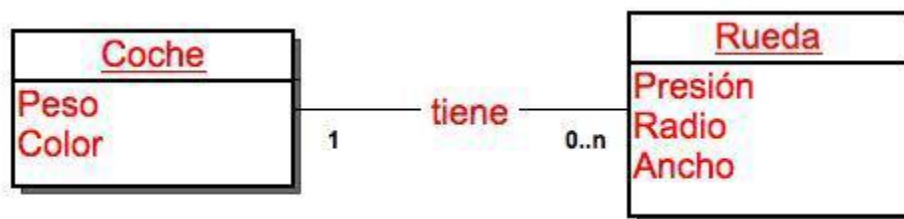


Figura 23. Ejemplo de UML

5.2.2 Modelo Relacional

Se especifica el modelo relacional de la aplicación utilizando UML, así podremos observar la estructura de la información.

5.2.2.1 Diagrama UML de la aplicación

Se definen las clases necesarias, así como las relaciones que mantienen y el número de relaciones.

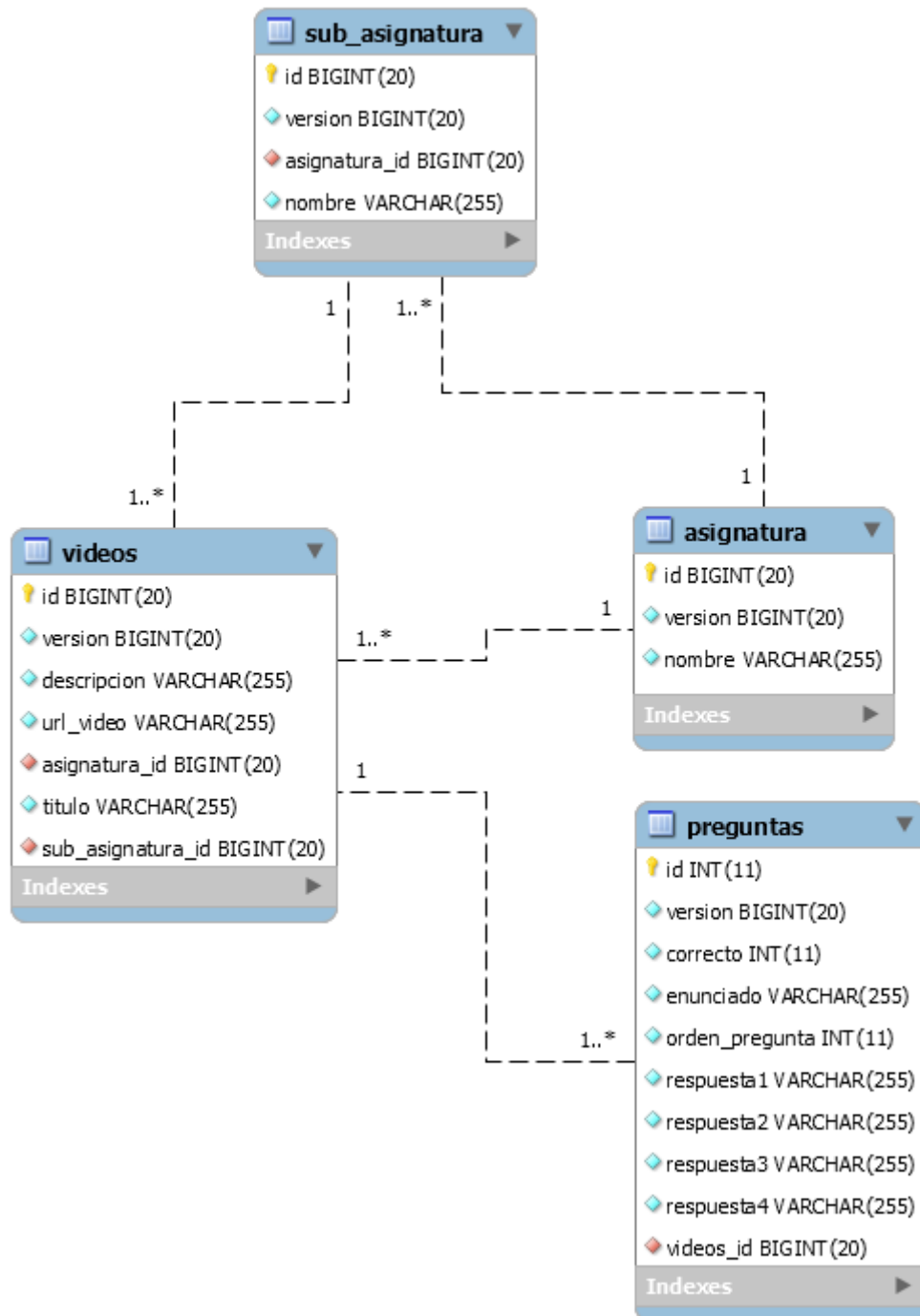


Figura 24. UML - Modelo Relacional

Se detallan las clases de manera independiente:

- **Clase Asignatura**

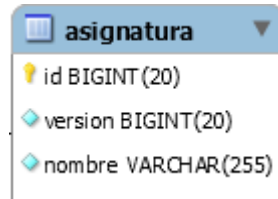


Figura 25. UML Asignaturas

Representa las asignaturas que se imparten en las aulas de clase, sus atributos son un identificador y un nombre de asignatura.

- **Clase SubAsignatura**

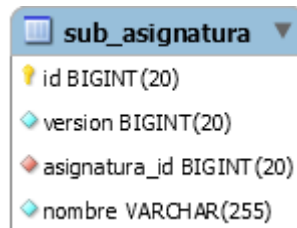


Figura 26. UML SubAsignaturas

De cada una de las Asignaturas podemos tener de una a varias subasignaturas, por esta razón hemos creado esta clase con los siguientes atributos: identificador de la subasignatura, nombre de la subasignatura y como hay relación de dependencia con asignatura tiene como clave ajena el id de asignatura.

- **Clase Videos**

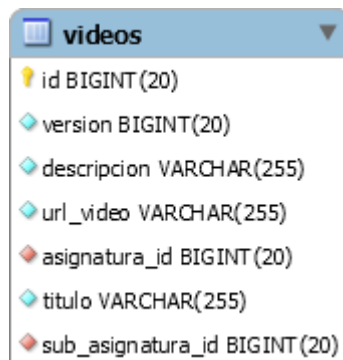


Figura 27. UML Videos

Esta tabla simbolizan a los videos almacenados en la aplicación con sus respectivos atributos, en este caso un identificador, una descripción, url del video, un título y como cada video es clasificado por asignatura y esta a su vez tiene subasignaturas, por lo tanto hay una relación de dependencia con asignaturas y subasignaturas manteniendo como clave ajena sus respectivos Identificadores de cada una.

- **Clase Preguntas**

Representa las preguntas que van asociadas a los videos y son las que aparecen automáticamente interrumpiendo la reproducción normal. Presentan los siguientes atributos: identificador, enunciado, orden pregunta, correcto, respuesta1, respuesta2, respuesta3 y respuesta4. Al tener que estar asociados con los videos se establece una relación de dependencia por lo tanto presenta como clave ajena el identificador del video.

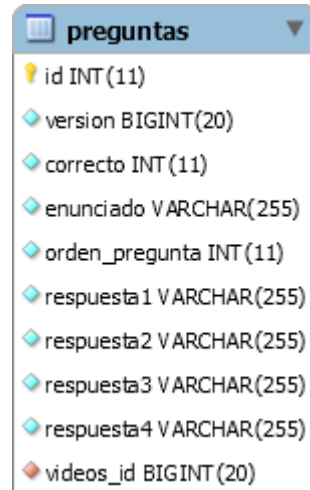


Figura 28. UML Preguntas

5.3 Vistas de la aplicación

Especificaremos cual serán las diferentes interfaces que se han desarrollado para los usuarios que hacen uso del sistema, mediante esta especificación se ayudará a conocer la estructura de la parte visual.

Elementos que componen las interfaces diseñadas son:

- **Imagen:** logo de la aplicación, en este caso logo de AprenvId (Aprendizaje mediante Videos).



Figura 29. Leyenda Imagen

- **Texto:** texto presente en la aplicación.

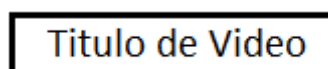


Figura 30. Leyenda Texto

- **Botones:** botones necesarios para enviar datos a la aplicación.

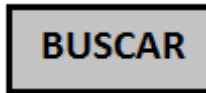


Figura 31. Leyenda Botones

- **Campo para introducir texto:** campo a introducir los datos solicitados.



Figura 32. Leyenda Introducir Texto

- **Video Seleccionado:** video seleccionado a reproducir.

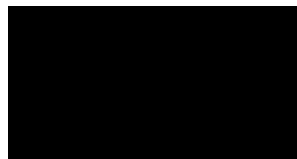


Figura 33. Leyenda Video

- **Enlace:** enlaces presentes en la aplicación.

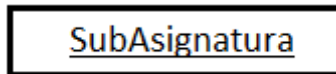


Figura 34. Leyenda Enlace

- **Campo Editable:** campos editables en la aplicación.



Figura 35. Leyenda Campo Editable

- **Campo de Selección:** elección a elegir entre las opciones de una lista desplegable o con radio buttons.

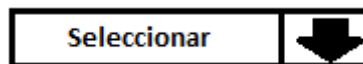


Figura 36. Leyenda Desplegable

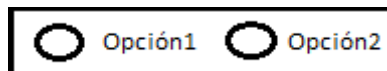


Figura 37. Leyenda Selección Radio Buttons

5.3.1 Interfaces Comunes

5.3.1.1 Pantalla Principal

Pantalla Principal de la aplicación, es común tanto para el Usuario Consumidor, quien hace uso de los videos y el Usuario Colaborador, quien si lo desea puede llegar a colaborar con AprenVid cediéndonos sus videos. En esta pantalla es posible hacer una búsqueda de Videos rellenando el campo editable con una palabra clave, o a través del catálogo de asignaturas.

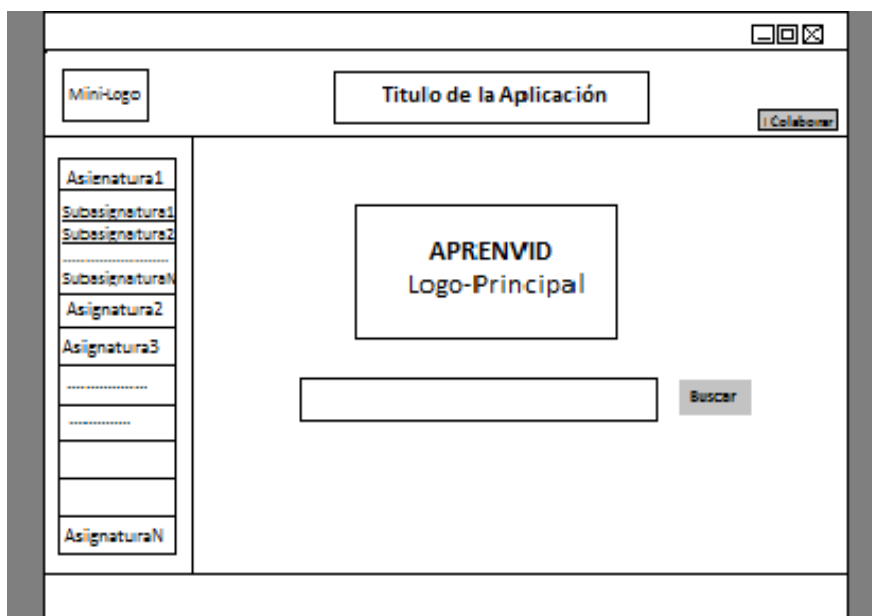


Figura 38. Página Principal Aplicación

5.3.1.2 Listado de Búsqueda General

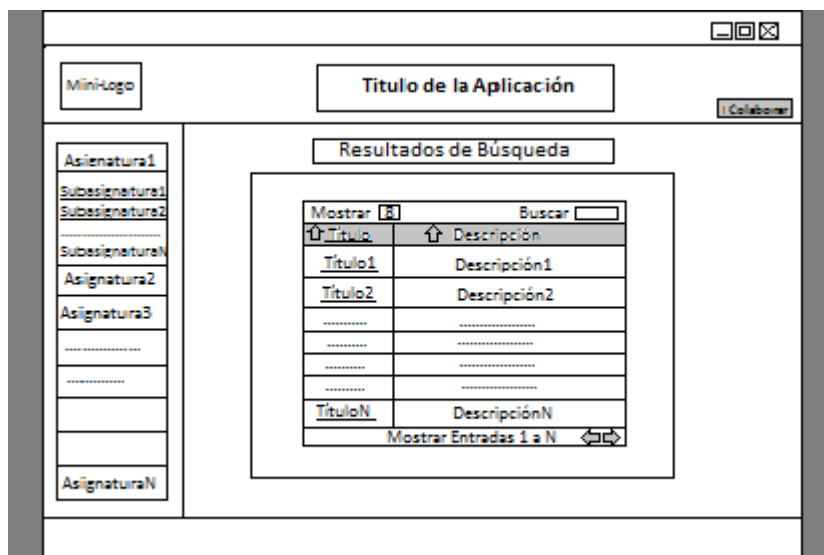


Figura 39. Listado de Búsqueda General

Una vez haya introducido el término general de búsqueda y pulse el botón Buscar el programa realizará una consulta buscando los videos que coincidan con la palabra de clave o en el título, o en la descripción, o en la asignatura o en la subasignatura a la que esté asociada el video.

Los resultados que coincidan con la consulta se mostrarán en una tabla que contiene el título y la descripción de los videos. En esta tabla el usuario tiene la posibilidad de ordenar alfabéticamente la información tanto por título o descripción, además de realizar una búsqueda dentro de la misma tabla y por último los resultados obtenidos son paginados.

5.3.1.3 Listado de Búsqueda por SubAsignaturas

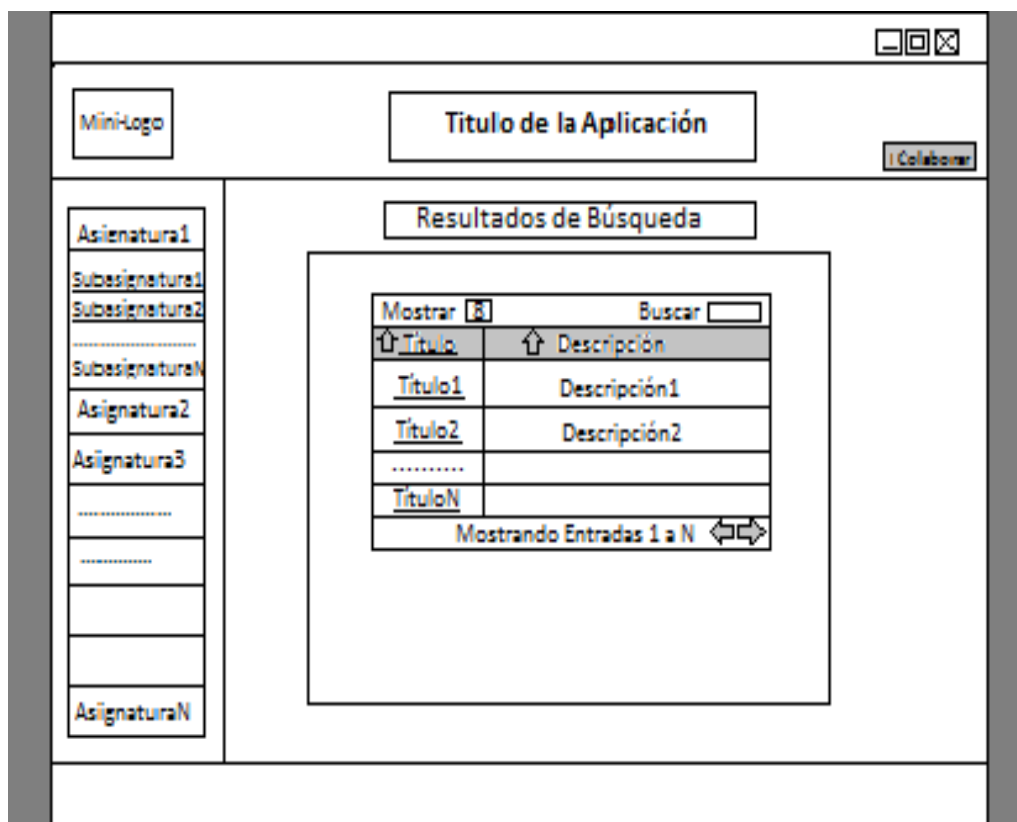


Figura 40. Listado de Búsqueda por SubAsignaturas

Esta vista es igual a la anterior, pero difieren en la forma de proceder, es decir el usuario pulsa el enlace de la asignatura de la que está interesado y a continuación se despliega todas las subasignaturas que forman parte de ella, de esta forma se presenta al usuario una categorización de los videos que tenemos disponibles.

En la Figura 40 se observa como aparecen las subasignaturas que forman parte de la Asignatura1.

Una vez seleccionada la subasignatura se obtiene la tabla de resultados, esta lista de resultados está más acotada, ya que solamente se mostrarán los videos que pertenecen a una determinada subasignatura.

Como en la anterior vista esta tabla permite ordenar alfabéticamente la información tanto por título o por descripción, además de realizar una búsqueda dentro de la misma tabla y por último los resultados obtenidos son paginados.

5.3.1.4 Reproducción de Video

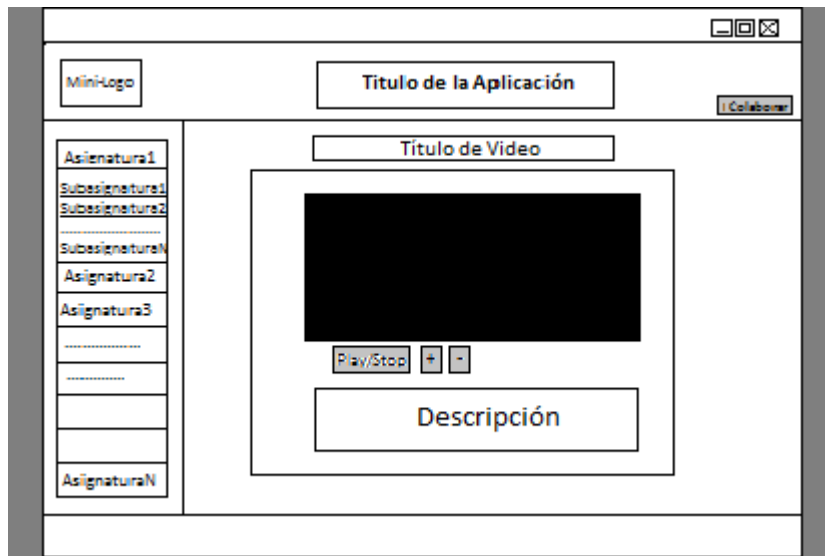


Figura 41. Reproducción del Video

Una vez hayamos obtenido los resultados de la consulta, seleccionaremos el video y acto seguido visualizaremos la imagen que se muestra, en esta página se dispone solamente de los botones "Play/Stop" que permite reproducir y parar el video, "+", permite aumentar el volumen y "-" permite disminuirlo. Los botones de avanzar o retroceder la reproducción no están disponibles para los usuarios, dado que solo se puede continuar y retroceder una vez se haya contestado correcta o incorrectamente las preguntas que aparecerán automáticamente, cabe destacar que la función de retroceder quiere decir repetir tramo del video segundo después del que se contesto correctamente una pregunta, no sé da opción al usuario de retroceder al momento que el desee y la función continuar es seguir con la reproducción normal del video, no existe la posibilidad de que el usuario avance al momento de reproducción que el desee.

5.3.1.5 Preguntas del Video

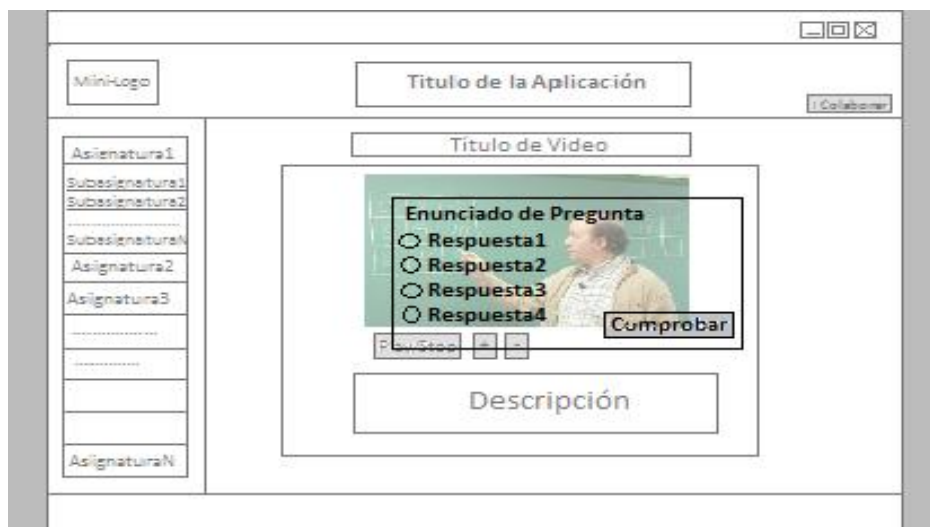


Figura 42. Ventana de Preguntas

La reproducción del video se verá interrumpida automáticamente por la aparición de una serie de preguntas que se dividirán en ventanas emergentes. La ventana emergente contiene el enunciado de la pregunta y las posibles respuestas, nuestra tarea es elegir una de las respuestas y pulsar el botón "Comprobar".

Una vez pulsado el botón comprobar, la siguiente ventana que se visualiza es la que se observa en la Figura 43, en esta ventana aparece un mensaje señalando si la respuesta ha sido contestada de forma correcta o incorrecta. Para cerrar la ventana emergente es necesario pulsar el botón aceptar.



Figura 43. Comprobación de Respuesta

Al pulsar el botón aceptar volvemos a la página en la que se encuentra el video, en el texto que indica si la respuesta ha sido contestada de forma correcta e incorrecta nos informará además si continuamos con la reproducción normal del video o es necesario repetir el último tramo del video reproducido antes de aparecer esta última pregunta. Por ello al pulsar Play el video continuará o volverá a reproducir el último tramo.

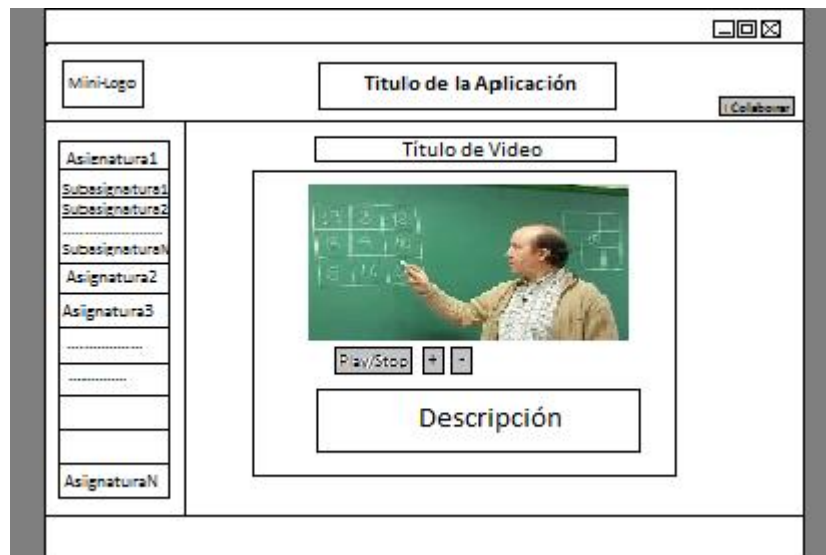


Figura 44. Volver a la Reproducción del Video

5.3.2 Interfaz de Usuario Colaborador

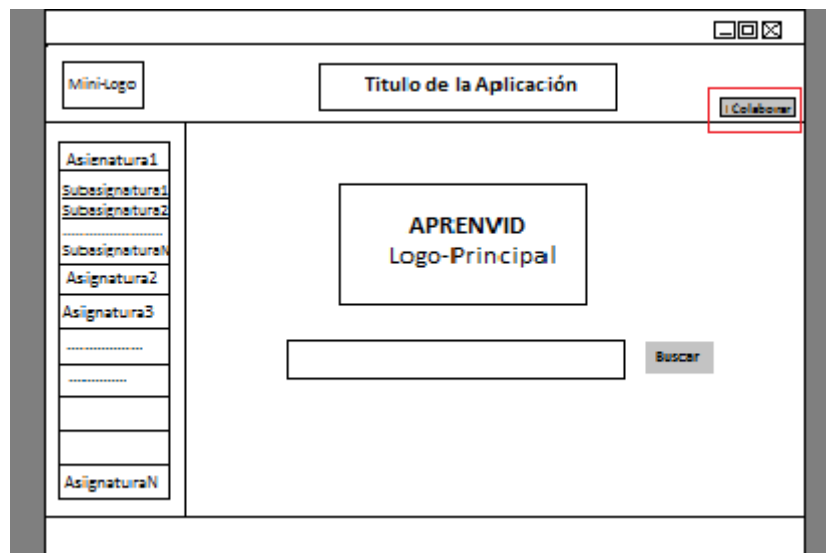


Figura 45. Página para el Usuario Colaborador

En esta ventana se puede observar que existe un botón Colaborar, este botón es usado por las personas que desean rellenar un formulario con el propósito de cedernos sus videos de aprendizaje y así estos formen parte de la aplicación.

Una vez pulsado el botón de Colaborar le aparecerá el siguiente Formulario de Colaboración.

Figura 46. Formulario de Colaboración

Una vez haya rellenado los datos del formulario se pulsa el botón enviar, en estos momentos AprenVid recibe un correo con la información facilitada, mientras que el usuario colaborador observa la siguiente pantalla.

Figura 47. Mensaje de Envío de Email

5.3.3 Interfaz de Administrador

5.3.3.1 Pantalla Principal

La página principal para el administrador es distinta a la que observa el usuario al que está destinada la aplicación, con la meta de aprender. En esta ventana el usuario tiene que acceder con su login y contraseña.

Diagrama de la Pantalla Principal de Administrador. La interfaz incluye un encabezado con un 'Mini-Logo' a la izquierda y el 'Titulo de la Aplicación' a la derecha. El contenido principal solicita al usuario que introduzca su usuario y contraseña. Se proporcionan dos campos de entrada: uno para el 'Usuario:' y otro para la 'Contraseña:'. Debajo de estos campos se encuentra un botón etiquetado como 'Acceder'.

Figura 48. Pantalla Principal de Administrador

5.3.3.2 Gestionar

Una vez el usuario administrador se haya identificado correctamente podrá acceder a la ventana donde tiene la posibilidad de Gestionar Asignaturas, Gestionar Subasignaturas, Gestionar Videos, Gestionar Preguntas.

Diagrama de la Pantalla Gestionar. La interfaz incluye un encabezado con un 'Mini-Logo' a la izquierda y el 'Titulo de la Aplicación' a la derecha. El contenido principal presenta cuatro botones de gestión dispuestos en una cuadrícula de dos columnas y dos filas: 'Gestionar Asignatura', 'Gestionar SubAsignaturas', 'Gestionar Videos' y 'Gestionar Preguntas'.

Figura 49. Pantalla Gestionar

5.3.3.2.1 Gestionar Asignaturas

- **Listado de Asignaturas**

Al pulsar el botón de Gestionar Asignaturas veremos un listado con todas y cada una de las asignaturas dadas de alta en nuestra base de datos.

Figura 50. Listado de Asignaturas

- **Alta de Asignatura**

Si deseamos dar de alta una nueva asignatura lo que debemos hacer es pinchar en el botón nueva asignatura, este botón estará disponible en las ventanas Listado de Asignatura, Consultar Asignatura y Editar Asignatura.

Figura 51. Alta de Asignaturas

La ventana que observamos es para proceder al Alta de una nueva Asignatura, rellenamos el campo nombre y pinchamos en el botón Crear, la asignatura será dada de alta, esto se

realizará de una forma correcta siempre y cuando el nombre que identifica la asignatura no haya estado previamente registrada en la base de datos, de lo contrario el sistema indicará un mensaje de error. Una vez creada la asignatura podemos asignarle las subasignaturas mediante el link que indica Agregar SubAsignaturas, e incluso se puede asignar ya los videos que estarán categorizados por esa asignatura pinchando en el enlace Agregar Videos.

- **Consulta de Asignatura**

Figura 52. Consulta de Asignaturas

En la vista de Lista de asignaturas si pinchamos en el nombre de la asignatura podemos ver en detalle todos sus datos, es decir el nombre con el que se identifica, los nombres de las subasignaturas y títulos de videos que estén asociados a ella.

- **Editar de Asignatura**

Figura 53. Editar Asignaturas

Para editar una asignatura es necesario primero consultar sus datos, en esta ventana está presente el botón editar, este botón se puede observar en la imagen "Consulta de Asignaturas", si pulsamos sobre él obtendremos la vista que se observa, una vez estemos en esta ventana es cuando podemos modificar todos sus campos: nombre, agregar subasignaturas o videos. Los datos se actualizará una vez se dé al botón actualizar.

- **Eliminar Asignatura**

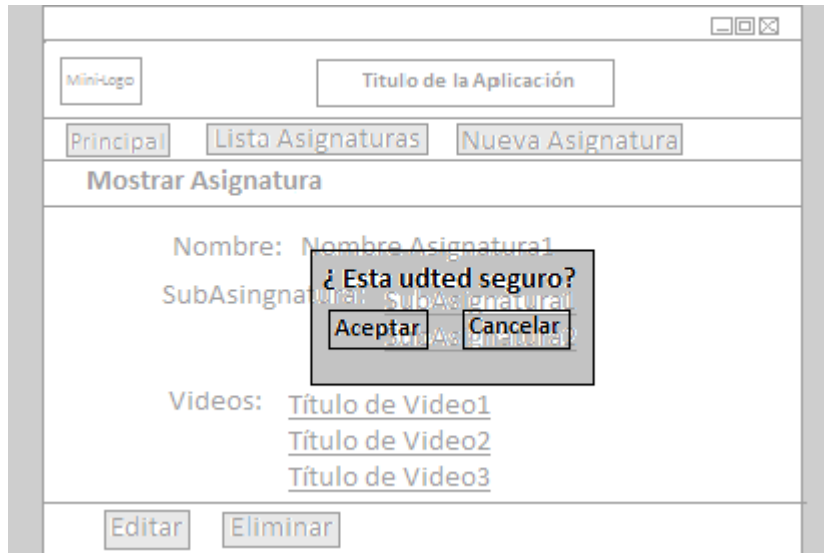


Figura 54. Eliminar Asignaturas

Así como para editar, se debe proceder con eliminar, es decir primero pinchamos sobre el Nombre de la asignatura disponible en el "Listado de Asignaturas", al pinchar sobre ella nos llevará a la consulta de sus datos y es en esta ventana donde está disponible el botón eliminar, se procede a pulsar este botón y acto seguido aparecerá una ventana emergente preguntándonos si estamos seguros de realizar esta acción. Esta ventana emergente tiene los botones Aceptar y Cancelar, en el caso de estar seguros pulsaremos el botón aceptar momento en el que la asignatura será eliminada siempre y cuando no tenga subasignaturas y videos asociados a ella.

5.3.3.2.2 Gestionar SubAsignaturas

- **Listado de SubAsignaturas**

Al pulsar el botón de Gestionar SubAsignaturas veremos un listado con todas y cada una de las Subasignaturas dadas de alta en la aplicación.

Figura 55. Listado de SubAsignaturas

- **Alta de SubAsignaturas**

Si deseamos dar de alta una nueva subasignatura lo que debemos hacer es pinchar en el botón nueva subasignatura, este botón estará disponible en las ventanas Listado de SubAsignatura, Consultar SubAsignatura y Editar SubAsignatura.

Figura 56. Alta de SubAsignaturas

Lo primero es seleccionar la Asignatura de la que formará parte, a continuación indicamos el nombre que identificará a la subasignatura, una vez relleno los datos solicitados, pinchamos en crear, con esto la subasignatura ha sido dada de alta, es necesario que el nombre que proporcionemos a la subasignatura no haya sido dado de alta previamente, en caso contrario el sistema indicará un error.

- **Consulta de SubAsignaturas**

Para consultar los datos de la subasignatura es necesario acceder al listado, una vez en esa ventana elegiremos la subasignatura a consultar, en ese momento en cuando mostrará sus datos en una nueva vista, tal y como se observa.

Figura 57. Consulta de SubAsignaturas

- **Editar SubAsignaturas**

Para editar la subasignatura lo primero es consultar los datos, en esa ventana existe un botón que indica editar, si lo pulsamos no llevará a la ventana en la que podemos editar sus campos, uno de ellos es cambiar el nombre de la Asignatura a la que esta agrupada, este campo es posible modificarlo solo por la que están registradas en la base de datos y el otro es cambiar el nombre que identifica a la subasignatura, tal y como se indicó en el Alta de SubAsignatura, el nombre elegido para identificarla no debe estar ya registrado en la base de datos, de lo contrario el sistema indicará este error.

Figura 58. Editar SubAsignaturas

- **Eliminar SubAsignaturas**

Para proceder a dar de baja una subasignatura presionamos el botón eliminar que está disponible en la ventana de consulta de subasignatura, una vez pulsado aparece una ventana emergente indicando si estamos seguros o no de realizar esta acción, si la subasignatura es la que deseamos eliminar, pulsamos el botón aceptar. La subasignatura se podrá dar de baja siempre que no haya videos asociados a ella.

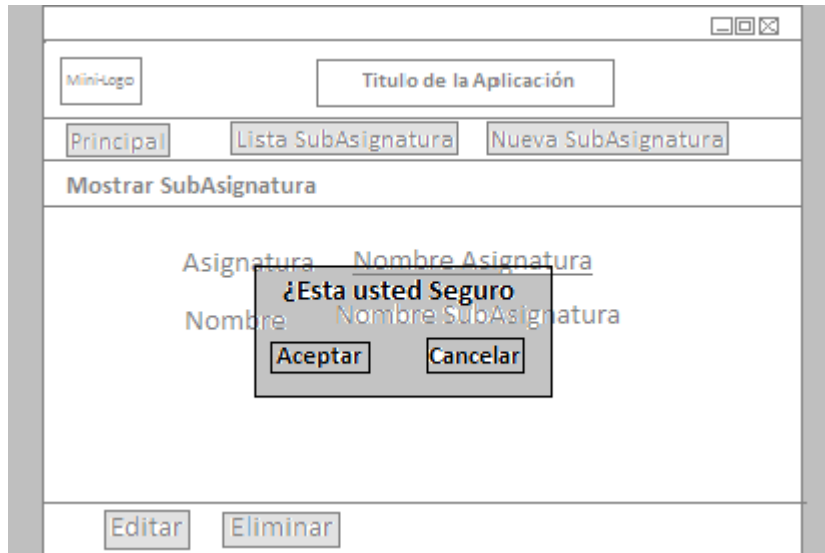


Figura 59. Eliminar SubAsignaturas

5.3.3.2.3 Gestionar Videos

- **Lista Videos**

Al pulsar el botón Gestionar Videos, divisaremos un listado con todos y cada uno de los videos dados de alta en el sistema.

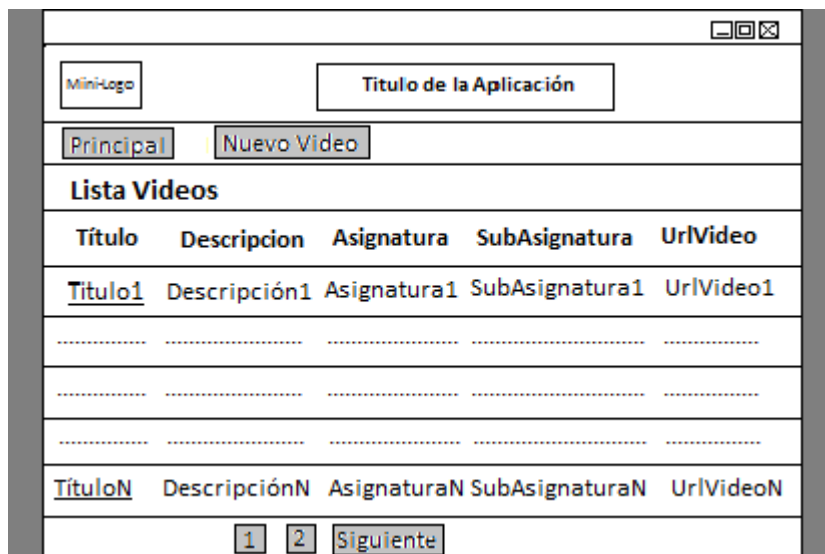


Figura 60. Listado de Videos

- **Alta Videos**

Para dar de alta un nuevo video en la base de datos debemos pulsar el botón Nuevo Video, este botón está disponible en las vistas de Listado Videos, Consulta Videos y Editar Videos.

Una vez pulsado el botón Nuevo Video observaremos la pantalla que se muestra a continuación, debemos rellenar los datos solicitados para proceder con el alta, las preguntas que irán asociadas al video pueden darse de alta desde esta ventana pulsando el botón Agregar Preguntas.

The screenshot shows a web application window with a title bar. Inside, there's a header with a 'Mini-Logo' and 'Titulo de la Aplicación'. Below that, a navigation bar contains 'Principal' and 'Lista de Videos'. The main content area is titled 'Crear Videos' and contains several input fields: 'Título', 'Descripcion', 'Asignatura' (with a dropdown arrow), 'SubAsignatura' (with a dropdown arrow), 'Preguntas' (with a link 'Agregar Preguntas'), and 'UrlVideos'. At the bottom, there is a 'Crear' button.

Figura 61. Alta de Videos

- **Consulta Videos**

Para consultar los datos de un video determinado es necesario acceder a su listado, una vez en el listado pulsaremos el enlace "Título del Video" en ese momento observaremos la vista Mostrar Videos, pantalla en la que se muestran todos los campos que identifican al video.

The screenshot shows a web application window with a title bar. Inside, there's a header with a 'Mini-Logo' and 'Titulo de la Aplicación'. Below that, a navigation bar contains 'Principal', 'Lista Videos', and 'Nuevo Video'. The main content area is titled 'Mostrar Videos' and contains a table with two columns. The left column lists fields: 'Título', 'Descripción', 'Asignatura', 'Preguntas', 'SubAsignatura', and 'UrlVideo'. The right column shows the corresponding data: 'Título de Video', 'Descripción de Video', 'Nombre de Asignatura', 'Enunciado Pregunta1', 'Enunciado Pregunta2', 'Enunciado Pregunta3', 'Nombre de .SubAsignatura', and 'Url Video'. At the bottom, there are 'Editar' and 'Eliminar' buttons.

Figura 62. Consulta de Videos

- **Editar Videos**

En la imagen anterior, se observa que al acceder a la consulta de un video está disponible el botón Editar, si pulsamos sobre él serán editables todos los campos que describen al video, una vez hayamos modificado el valor de los campos que deseamos corregir pulsaremos el botón actualizar para guardar los cambios.

Figura 63. Editar Videos

- **Eliminar Videos**

En la vista de consulta de un video está disponible el botón eliminar, si pulsamos sobre él aparece la ventana emergente indicando si estamos seguros de realizar esta acción, si pulsamos en aceptar el video será dado de baja siempre y cuando no tenga preguntas asociadas a él.

Figura 64. Eliminar Videos

5.3.3.2.2 Gestionar Preguntas

- **Listado de Preguntas**

Al pulsar el botón de Gestionar Preguntas obtendremos un listado con todas y cada una de las preguntas dadas de alta en el sistema.

Enunciado	OrdenPregunta	Correcto	Respuesta1	Respuesta2	Respuesta3	Respuesta4
EN1	OP1	C1	R1	R2	R3	R4
.....
.....
.....
ENN	OPN	CN	R1N	R2N	R3N	R4N

Figura 65. Listado de Preguntas

- **Alta Preguntas**

Si necesitamos dar de alta una pregunta que irá asociada a un video ya disponible en la base de datos debemos pinchar sobre el botón Nueva Pregunta que se encuentra en las ventanas de Listado Preguntas, Mostrar o Consultar Preguntas y en Editar Preguntas.

Se rellenar todos los datos indicados en la vista que se observa y se pincha sobre el botón Crear, con esto ya estaría disponible en la base de datos siempre que todos los datos hayan sido rellenados de forma correcta.

Figura 66. Alta de Preguntas

- **Consultar Preguntas**

Es posible consultar los datos que describen a una pregunta pinchando sobre el Enunciado de de Pregunta, este enlace está disponible en el listado general de Preguntas, una vez pulsado podremos ver la vista Mostrar Preguntas.

Figura 67. Consulta de Preguntas

- **Editar Pregunta**

Si algún dato introducido en la pregunta es incorrecto es posible editarlo mediante el botón editar que esta visible en la ventana de Mostrar Pregunta, una vez presionado el botón algunos campos de la pregunta elegida se pondrán de forma editable y otros de tal manera que solo puedan ser elegidos en un lista de selección. Se realizan los cambios oportunos y se pincha sobre el botón actualizar, en ese momento estará ya disponible el video con los nuevos datos.

Figura 68. Editar Preguntas

- **Eliminar Pregunta**

Al igual que en los casos anteriores es posible dar de baja las preguntas asociadas a un video, aparecerá la venta emergente tal y como se observa preguntándonos si estamos seguros de la operación a realizar, si es así, presionamos Aceptar, en caso contrario cancelamos la operación. Si presionamos sobre el botón Aceptar, no debería haber ningún tipo de problema, el sistema daría de baja a la pregunta elegida.

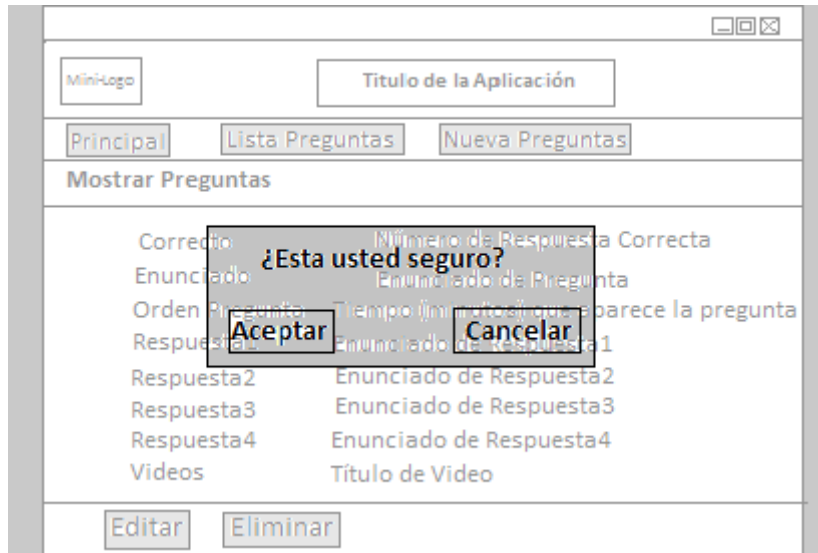


Figura 69. Eliminar Preguntas

5.4 Controladores de la aplicación

Para la parte de los controladores, la aplicación va a contar con uno por cada clase de dominio que gestionará todas las acciones a realizar para cada una de ellas y uno en especial que se encargará de acciones concretas.

- **Controlador de Gestión de Asignaturas:** se encarga de todas las acciones que conciernen a las asignaturas, entre las que están:
 - Creación de asignaturas
 - Edición de asignaturas
 - Consulta de asignaturas
 - Listado de asignaturas
 - Eliminación de asignaturas
- **Controlador de Gestión de SubAsignaturas:** se encargará de las acciones que tienen que ver con las subasignaturas, es decir:
 - Creación de subasignaturas
 - Edición de subasignaturas
 - Consulta de subasignaturas
 - Listado de subasignaturas
 - Eliminación de subasignaturas

- **Controlador de Gestión de Videos:** este controlador tiene el objetivo de realizar todas las acciones que corresponden a los videos, las enumeramos a continuación:
 - Creación de Videos
 - Edición de Videos
 - Consulta de Videos por parte del administrador
 - Listado de Videos
 - Eliminación de Videos
 - Consulta de Videos por parte del usuario, dependiendo de un criterio de búsqueda.
 - Obtener las preguntas asociadas al video determinado.
 - Comprobación de las respuestas dada a las preguntas asociadas al video.
 - Controla la secuencia del video una vez contestada una pregunta.

- **Controlador de Gestión de Preguntas:** controlador encargado de realizar las acciones que corresponden a las preguntas, las detallamos a continuación.
 - Creación de Preguntas
 - Edición de Preguntas
 - Consulta de Preguntas
 - Listado de Preguntas
 - Eliminación de Preguntas

- **Controlador Home:** se ha creado un controlador de acciones concretas que no pertenecen a los controladores anteriores, se listan las tareas que realiza:
 - Autoriza el acceso al administrador de la aplicación
 - En este controlador se accede a la página principal tanto del administrador como la del usuario.
 - Ofrece la vista al usuario administrador para la gestión de asignaturas, subasignaturas, videos y preguntas.
 - Proporciona la vista del formulario de colaboración.
 - Envía el email con los datos del usuario colaborador a los administradores de la aplicación.
 - Muestra la ventana de confirmación del envío del email al usuario colaborador.

CAPÍTULO 6

6. Implementación de la Aplicación

En este capítulo se detalla la implementación de las funcionalidades que se han realizado para el desarrollo de la aplicación.

6.1 Estructura de la aplicación

Los ficheros de código se encuentran organizados en la aplicación siguiendo la estructura de carpetas que se indican a continuación:

- **Carpeta *domain*:** en esta carpeta se almacenan todas aquellas clases que representa un objeto en la aplicación. En el caso que nos concierne podemos destacar la clase *Videos*.
- **Carpeta *controllers*:** en esta carpeta se sitúa los controladores del patrón MVC.
- **Carpeta *views*:** en esta carpeta se incluyen las vistas del patrón MVC, estas serán las diferentes interfaces que podrán ver los usuarios.
- **Carpeta *services*:** carpeta que contiene aquellos métodos que son utilizados o pueden ser utilizados por varios controladores, de tal forma que se externalizan y se puede acceder a ellos.
- **Carpeta *plugins*:** esta carpeta alberga los diferentes plugins de los que hemos hecho uso para el proyecto.
- **Carpeta *taglib*:** en esta carpeta se almacenan las diferentes librerías de etiquetas propias.
- **Carpeta *asset*:** desde las versiones 2.2 de Grails se está usando el plugins Asset-Pipeline que crea la carpeta *asset*, este plugins es utilizado para la gestión de recursos, facilita el procesamiento y compilación para los archivos CSS y JS. Por lo tanto es en esta carpeta donde podemos encontrar las hojas de estilo (ficheros *.css*), las imágenes utilizadas en la aplicación y ficheros Javascript (con extensión *.js*). En las versiones anteriores a las versiones 2.2 se utilizaba para este mismo fin la carpeta ***web-app***, de momento se mantiene esta carpeta pero ya no se utiliza.
- **Carpeta *test*:** en esta carpeta se ubican los ficheros de las diferentes pruebas que se realiza en la aplicación.

A continuación se presenta la imagen en la que se puede observar y nos permite hacernos una idea de cómo se estructura las carpetas antes descritas en el proyecto.

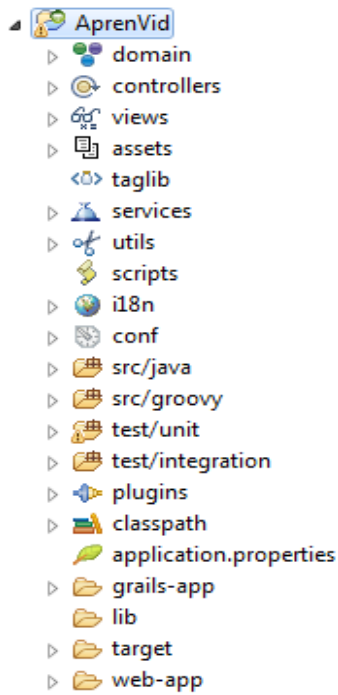


Figura 70. Estructura de la Aplicación

6.2 Código de la aplicación

Tal y como se indicó anteriormente la aplicación se estructura por carpetas, en las que cada una de ellas contienen ficheros del mismo tipo.

En cada una de estas carpetas, los archivos que las conforman están organizados dentro de un paquete denominado ***es.proyecto.aprenvid***.

6.2.1 Clases de dominio

Las clases que se explicaron en el modelo de datos, son las clases de dominio que necesita la aplicación, cada una se corresponde con la representación de los objetos necesarios de este trabajo. Las clases de dominio contendrán atributos y métodos específicos para cada una de ellas.


```

package es.proyecto.aprenvid

class Videos {

    String urlVideo
    String descripcion
    String titulo

    static hasMany = [preguntas:Preguntas]
    static belongsTo = [asignatura:Asignatura,subAsignatura:SubAsignatura]

    static constraints = {
        titulo(blank:false,size:3..50)
        descripcion(blank:false,maxSize:999999)
        urlVideo(blank:false)
    }

    String toString(){
        return titulo
    }
}

```

Figura 71. Clase de Dominio Videos

En la imagen anterior se observa varios elementos que pertenecen a una clase de dominio:

- La primera línea que se visualiza indica el paquete donde se encuentra situada la clase.
- A continuación la definición de la clase, entre el paquete y la definición de las clases irían las clases importadas si fueran necesarias.
- Seguidamente esta los atributos, titulo, descripción y urlVideo. En Groovy cuando una clase tiene relación de dependencia con otra, se establecen atributos especiales, en este caso tenemos el atributo *hasMany* que representa una relación 1..N, en el lado del objeto único que en este caso es Videos tenemos *hasMany* y en el extremo N, que son las preguntas, tendría el atributo *belongsTo* presente en la clase de dominio Preguntas, aunque precisamente en el caso de la Figura 71 tenemos presente el atributo *belongsTo*, con extremo N, sería nuestra clase Videos, mientras que las Asignaturas y SubAsignaturas sería el extremo 1, por lo que tendría como atributo en cada una de ellas *hasMany*.
- Los *constraints* representan las opciones de validación del objeto. Por defecto un atributo no puede ser nulo, pero en ocasiones conviene tener un atributo nulo, para ello basta con introducir *nullable: true*, en nuestro caso necesitamos que los atributos sean no nulo. Además se ha especificado que titulo tenga un valor mínimo de 3 caracteres y máximo de 50, en cuanto a la descripción no se ha establecido un mínimo pero sí un máximo de 99999, dado que las descripciones suelen ser comúnmente largas, se ha establecido ese límite. Otros aspectos que se pueden controlar con los *constraints* es permitir valores en blanco o que sean únicos, o que los atributos estén dentro de un rango establecido. Tal y como se observa con *constraints* se pueden controlar varios aspectos importantes de los atributos.

- En este caso no está presente *mapping* dado que no necesitamos hacer correspondencia de atributos de las clases con los nombres de columna en una base de datos ya creada, si partiéramos de una base de datos ya creada con anterioridad a la aplicación es preciso especificar aquí cual es el atributo que actuará de identificador en la tabla de la base de datos.
- Por último es necesario especificar un método que devuelve el título del video ya que por el contrario nos devolvería el programa automáticamente el nombre de la clase unido al nombre del paquete, por lo tanto este método es necesario crearlo en todas las clases para que nos devuelva la información que nosotros necesitamos.
- Algo muy importante en Groovy y que facilitan las operaciones para obtener o asignar valores de los atributos, los denominados *getter* o *setter* se omiten, salvo que si se desea sobrescribir la funcionalidad por defecto, en cuyo caso solo hace falta escribir el método correspondiente.

6.2.2 Controladores

En el capítulo de diseño se han indicado los controladores que se han creado y las operaciones que realiza cada uno. Vamos a detallar los que necesitamos e incluiremos una imagen para observar su aspecto visual.

- Controlador Home: los métodos que la conforman tiene como objeto mostrar las vistas principales de la aplicación, además de permitir el uso del formulario de colaboración para las personas que solicitan rellenarlo, por otro lado controla el acceso del administrador y el envío del email con los datos de los usuarios colaboradores a los administradores de la aplicación. Hay que especificar que este controlador para llevar a cabo el envío del email hace uso del servicio *homeService* en este servicio tenemos un único método que hace el envío del email con los datos del formulario y como medida de control para cuando se rellena un formulario el uso del servicio *simpleCaptchaService* que genera imágenes Captcha para el formulario, para el uso de este servicio es necesario el plugins *simpleCaptcha*.
- Controlador Asignaturas: se encarga de listar todas las asignaturas, mostrar en detalle a cada una de ellas, además de permitir dar de alta, eliminar o modificar las asignaturas ya registradas en el sistema. Este controlador tiene además la misión de que cada vez que se quiera dar de alta un video categorizándolo para una asignatura y subasignatura, cuando seleccionemos una asignatura de la lista desplegable te cargue automáticamente la lista desplegable de las subasignaturas que pertenecen a la asignatura elegida, es decir la misión es crear campos select dinámicos
- Controlador SubAsignaturas: con este controlador paso lo mismo que las Asignaturas, muestra la lista de subasignaturas, permite la consulta de una subasignatura en detalle, además de las acciones de alta, baja y modificación.
- Controlador Video: Este controlador contiene las operaciones de listar, consultar, crear, eliminar y modificar los videos dados de alta en el sistemas, además de esto tiene la tarea de consultar los videos según los criterios de búsqueda general y búsqueda por subasignaturas, para esta tarea hace uso de un solo método que maneja las dos consultas, por otro lado conviene destacar que está encargado del obtener las preguntas de un video determinado, que aparecerán interrumpiendo la reproducción normal del video, en los tiempos establecidos, además tiene como tarea comprobar

las respuestas dadas por el usuario y por último controlar la secuencia que debe seguir el video según las respuestas dadas. Para la labor de obtener las preguntas y la de controlar la secuencia del video hace uso del servicio PreguntasService.

- Controlador Pregunta: la labor de este controlador es listar, consultar, crear, eliminar y modificar las preguntas dadas de alta en el sistema por el administrador.

A continuación vamos a explicar un ejemplo ilustrativo de unos de los controladores más importantes de la aplicación, es el caso del controlador Videos:

```
package es.proyecto.aprenvid
import static org.springframework.http.HttpStatus.*
import grails.transaction.Transactional

@Transactional(readOnly = true)
class VideosController {
    def preguntasService

    static allowedMethods = [save: "POST", update: "PUT", delete: "DELETE"]

    def index(Integer max) {
        params.max = Math.min(max ?: 10, 100)
        respond Videos.list(params), model:[videosInstanceCount: Videos.count()]
    }

    def show(Videos videosInstance) {
        respond videosInstance
    }

    def create() {
        respond new Videos(params)
    }
}
```

Figura 72. Controlador Videos - Parte1

```

def save(Videos videosInstance) {
    if (videosInstance == null) {
        notFound()
        return
    }

    if (videosInstance.hasErrors()) {
        respond videosInstance.errors, view: 'create'
        return
    }

    videosInstance.save flush: true

    request.withFormat {
        form multipartForm {
            flash.message = message(code: 'default.created.message',
                                   args: [message(code: 'videos.label',
                                                    default: 'Videos'), videosInstance.id])
            redirect videosInstance
        }
        '*' { respond videosInstance, [status: CREATED] }
    }
}

def edit(Videos videosInstance) {
    println videosInstance.asignatura
    respond videosInstance
}

def update(Videos videosInstance) {
    if (videosInstance == null) {
        notFound()
        return
    }

    if (videosInstance.hasErrors()) {
        respond videosInstance.errors, view: 'edit'
        return
    }

    videosInstance.save flush: true

    request.withFormat {
        form multipartForm {
            flash.message = message(code: 'default.updated.message',
                                   args: [message(code: 'Videos.label', default: 'Videos'),
                                           videosInstance.id])
            redirect videosInstance
        }
        '*' { respond videosInstance, [status: OK] }
    }
}

```

Figura 73. Controlador Videos - Parte2

```

def delete(Videos videosInstance) {

  if (videosInstance == null) {
    notFound()
    return
  }

  videosInstance.delete flush:true

  request.withFormat {
    form multipartForm {
      flash.message = message(code: 'default.deleted.message',
        args: [message(code: 'Videos.label', default: 'Videos'),
          videosInstance.id])
      redirect action:"index", method:"GET"
    }
    '*'{ render status: NO_CONTENT }
  }
}

protected void notFound() {
  request.withFormat {
    form multipartForm {
      flash.message = message(code: 'default.not.found.message',
        args: [message(code: 'videos.label', default: 'Videos'), params.id])
      redirect action: "index", method: "GET"
    }
    '*'{ render status: NOT_FOUND }
  }
}

```

Figura 74. Controlador Videos - Parte3

```

def resultadoBusquedaPrincipal(){
  def buscarGeneral = params.buscar
  def buscarPorSubAsignaturas=params.idSA
  def listBusqueda
  def asignaturaB
  def intanciaSubAsignatura

  if(buscarGeneral){

    asignaturaB = Asignatura.findByName(buscarGeneral)
    def c = Videos.createCriteria()

    listBusqueda = c{

      if(asignaturaB){

        or{
          eq("asignatura",asignaturaB)
          like("titulo",""+buscarGeneral+"")
          like("descripcion",""+buscarGeneral+"")
        }
      }else{
        or{
          like("titulo",""+buscarGeneral+"")
          like("descripcion",""+buscarGeneral+"")
        }
      }
    }
  }

  if(buscarPorSubAsignaturas){
    intanciaSubAsignatura= SubAsignatura.get(buscarPorSubAsignaturas)
    listBusqueda = Videos.findAllWhere(subAsignatura:intanciaSubAsignatura)
  }
  return [listBusqueda:listBusqueda]
}

```

Figura 75. Controlador Videos - Parte4

```

def showVideo(){
  String ordenPregunta, segundosTotalesVideo
  def videoBuscado
  if(params.idd){
    videoBuscado = Videos.get(params.idd)
    ordenPregunta= preguntasService.obtenerPrimeraPregunta(videoBuscado)

    return [videoBuscado:videoBuscado,ordenPregunta:ordenPregunta]
  }else{
    render(template:"templates/divShowVideo",
           model:[videoBuscado:videoBuscado,
                  ordenPregunta:ordenPregunta])
  }
}

def obtenerPregunta(){
  def instanciaPregunta
  def instanciaVideo
  def listPreguntas
  int orden
  if(params.idd && params.tiempoP){
    orden = Integer.parseInt(params.tiempoP)
    instanciaVideo = Videos.get(params.idd)
    listPreguntas=Preguntas.findAllWhere(videos:instanciaVideo)
    listPreguntas.sort{it.ordenPregunta}.each {
      println it.ordenPregunta
      if(it.ordenPregunta == orden){
        instanciaPregunta=it
      }
    }
  }

  render(template:"templates/divTestPreguntas",
        model:[instanciaPregunta:instanciaPregunta])
}

```

Figura 76. Controlador Videos - Parte5

```

def comprobarRespuesta(){
  def instanciaPregunta
  int numRespuesta
  String cadena,ok
  if(params.idd && params.respuesta){
    instanciaPregunta=Preguntas.get(params.idd)
    cadena = params.respuesta
    numRespuesta= Integer.parseInt(
      cadena.substring(cadena.length()-1,cadena.length()))
    if(instanciaPregunta.correcto==numRespuesta){
      ok="verdadero"
    }else{
      ok="falso"
    }
  }
  render(template:"templates/divTestPreguntas",
    model:[ok:ok,instanciaPregunta:instanciaPregunta])
}

def comprobarSecuenciaVideo(){
  Preguntas preguntaActual,anteriorPregunta
  def listPreguntas
  Videos instanciaVideo
  String ordenPregunta
  String accion,tiempoInicio
  String tiempoPrimeraPregunta
  if(params.iddV && params.iddP && params.ok){
    preguntaActual = Preguntas.get(params.iddP)
    instanciaVideo = Videos.get(params.iddV)
    listPreguntas = instanciaVideo.preguntas
    tiempoInicio=preguntaActual.ordenPregunta.toString()
    if(params.ok == "verdadero"){
      ordenPregunta = preguntasService.siguientePregunta(preguntaActual,listPreguntas)
      tiempoInicio=preguntaActual.ordenPregunta.toString()
      accion = "avanzar"
    }else if(params.ok == "falso"){
      tiempoInicio=preguntasService.anteriorPregunta(preguntaActual,listPreguntas)
      ordenPregunta=preguntaActual.ordenPregunta.toString()
      accion = "retroceder"
    }
    if(ordenPregunta=="-1"){
      tiempoPrimeraPregunta= preguntasService.obtenerPrimeraPregunta(instanciaVideo)
    }else{
      tiempoPrimeraPregunta = "0"
    }
  }

  render(template:"templates/divShowVideo",
    model:[videoBuscado:instanciaVideo,
      ordenPregunta:ordenPregunta,accion:accion,
      tiempoInicio:tiempoInicio,
      tiempoPrimeraPregunta:tiempoPrimeraPregunta])
}

```

Figura 77. Controlador Videos - Parte6

En las primeras líneas podemos encontrar lo que se comentó anteriormente, es decir el paquete donde se encuentra el fichero, inmediatamente después las librerías que necesita importar y la definición de la clase, acto seguido se observa la declaración del servicio que necesita el controlador. La siguiente línea declara un variable estática *allowedMethods*, esta

variable ofrece una sintaxis declarativa sencilla para especificar qué métodos HTTP se permiten para las acciones del controlador, por defecto todos los métodos de petición se permite para todas las acciones del controlador. La propiedad `allowedMethods` es opcional y sólo es necesario definir si el controlador tiene acciones a ser restringidas a ciertos métodos de petición, como es este caso, solo es permitido el método `Http POST` (indica al servidor que se prepare para recibir información del cliente, se usa para enviar información de formularios) para el método `Save` del controlador, el método `Http PUT` (envía el recurso identificado en la URL desde el cliente hacia el servidor) para el método `Update` y por último el método `Http DELETE` (solicita al servidor que borre el recurso identificado en la URL) para el método `Delete` del controlador.

Vamos a proceder a explicar los métodos que conforman el controlador. Conviene destacar que en Groovy la política por defecto es que todos los objetos, métodos y atributos son públicos a menos que se especifique lo contrario. El controlador contiene los métodos *create*, *resultadoBusquedaPrincipal*, *showVideo*, *obtenerPregunta*, *comprobarRespuesta*, *comprobarSecuenciaVideo* que devuelven un valor, este puede ser nulo o contener un objeto. Además estos métodos no reciben parámetros.

Una vez se declaran los métodos y estos con públicos, son accesibles desde cualquier otra vista, servicio o controlador que lo llame.

Los métodos `show`, `create`, `save`, `edit`, `update`, `delete` y `notFound` son comunes para todos los controladores, excepto para *Home*.

Detallamos las operaciones realizadas en cada uno de los métodos,

- **Método Index:** se encarga de darnos la lista de todos los videos dados de alta en el sistema, por defecto en cada uno de los controladores de los que disponemos hay un método `index`, en el caso del controlador *Home* solo da como resultado la vista principal pero en el resto de controladores este método realiza la operación *Clase.list()*, en este caso *Video.list()*, para el resto *Asignaturas.list()*, *SubAsignatura.list()* y *Preguntas.list()*, cada uno de ellos recibe un parámetro que se corresponde con el número de instancias a mostrar en la vista actual.
- **Método Show:** método encargado de mostrar el video que hayamos seleccionado de la lista que proporciona el método `index`, se aplica la misma operación para los controladores *Asignaturas*, *SubAsignaturas* y *Preguntas*.
- **Método Create:** su misión es mostrar el formulario de alta de un video cuando hayamos pulsado sobre el botón nuevo video. Los controladores *Asignaturas*, *SubAsignaturas* y *Preguntas* también disponen de este método que en cada uno de los casos muestra un formulario con los atributos que le correspondan.
- **Método Save:** una vez se haya rellenado el formulario que da el método `Create`, se procede a pulsar el botón `Crear` que en realidad está enviando los datos del formulario al método `save` para proceder con el alta en la base de datos. En este método se comprueba que los datos enviados cumplen las condiciones establecidas en los `constraints`, por este motivo si encuentra algún error nos devuelve al formulario de alta indicándonos el tipo de error encontrado. Si todo ha ido de forma correcta mostrara la vista que proporciona el método *show* con un mensaje que indica video ha

sido registrado. Esta misma explicación es perfectamente válida para el resto de controladores que disponen de este tipo de método, como es el caso de Asignaturas, SubAsignaturas y Preguntas

- Método Edit: al pulsar el botón editar disponible en la ventana de mostrar los datos del video hace la llamada a este método para devolver la vista del formulario con los datos de los video a modificar. El mismo comportamiento para Asignatura, SubAsignatura y Preguntas.
- Método Update: cuando ya disponemos de la vista con el formulario relleno con los datos del video a modificar, se realizan los cambios que necesitamos y procedemos a pulsar el botón actualizar, una vez pulsado llama a este método para que salve los nuevos datos proporcionados, si todo ha ido bien nos devuelve a la vista de mostrar datos del video con un mensaje indicando que los cambios se ha realizado de forma correcta, si algo ha ido mal nos devuelve a la vista del formulario con el error producido. Esta explicación es perfectamente válida para el resto de controladores Asignatura, SubAsignatura y Preguntas.
- Método Delete: a este método le llegan las peticiones hechas al pulsar el botón eliminar disponible en la vista de mostrar videos, se procederá a dar de baja el video en el caso de no tener Preguntas asociadas, si tuviese alguna pregunta, no sería posible darlo de baja hasta que no se hayan eliminado todas sus preguntas. Para el resto de casos: las asignaturas solo podrás ser eliminadas siempre que no contengan subasignaturas y videos asociada, para las subasignaturas no se podrán eliminar si tienen videos asociados, las únicas instancias que se pueden eliminar sin ningún tipo de restricción de dependencias son las preguntas.
- Método NotFound: este método es usado por los métodos *save*, *update* y *delete* del controlador Video, la labor de este método es controlar las peticiones que llegan de forma incorrecta a estos métodos, podemos citar un ejemplo como el solicitar borrar un video que no haya sido dado de alta, o en el caso de llegar alguna instancia Null para este tipo de peticiones. La misma misión para asignaturas, subasignaturas y preguntas.
- Método resultadoBusquedaPrincipal: a este método le llegan las peticiones de búsqueda por un término general y por subasignaturas.
 - Para realizar la búsqueda por un término general se crea una consulta Criteria. Cabe indicar que un criteria es un *dynamic finders* (localizadores dinámicos), tiene el mismo aspecto que los métodos estáticos, pero no existen en realidad. Cuando son invocados la primera vez se generan dinámicamente mediante la síntesis de bytewcodes.
Se ha planteado usar un criteria ya es necesario hacer una búsqueda avanzada que incluye muchos campos y comparaciones complejas, en este caso debemos buscar tanto en los campos títulos, descripción, asignatura y subasignatura el término de búsqueda, solo es necesario que el término esté presente en alguno de los campos. Una vez se realice la consulta devuelve la lista de resultados obtenidos.
 - La búsqueda por subasignaturas es más sencilla por este motivo se hace uso de un *dynamic Finders*, es un método compuesto por el prefijo *finAll* combinado don la operación *Where*, uniéndolo se convierte en *findAllWhere*,

en este método es necesario pasa como parámetro el nombre del atributo seguido de dos puntos y el valor de la búsqueda, es decir para el caso que nos atañe buscar, pasaríamos el nombre del atributo `subAsignatura: "valor de subasignatura"`, el atributo `subAsignatura` pertenece a la clase de dominio `Videos`. Quedaría de la siguiente manera `Videos.findAllWhere (subAsignatura: "valor a buscar")`. Una vez se proceda con la consulta devuelve la lista de resultados obtenidos.

- Método `showVideo`: mediante este método se puede visualizar el video solicitado, una vez haya sido elegido de la lista de resultados que proporciona una consulta. Este método también tiene la tarea de obtener el tiempo en el que debe aparecer la primera pregunta dado que es necesario conocer de antemano la primera pausa que se realizará de forma automática al video.
- Método `obtenerPregunta`: este método está encargado de consultar en los momentos establecidos las preguntas del video para mostrarlas en la ventana emergente y el usuario procede a resolverla.
- Método `comprobarRespuesta`: método encargado de verificar si la respuesta proporcionada por el usuario es correcta o incorrecta, este hecho se notifica al usuario con un mensaje en la ventana emergente.
- Método `comprobarSecuenciaVideo`: a este método se le ha delegado la tarea de comprobar la respuesta proporcionada por el usuario, si la secuencia del video continua desde el punto que se interrumpió la reproducción para visualizar la pregunta, quiere decir que el usuario contesto correctamente, si fuera el caso de haber contestado de forma incorrecta se deberá volver a reproducir el tramo de video antes de que aparezca la pregunta.

6.2.3 Vistas

Las vistas son los ficheros que permiten mostrar la información e interactuar con los usuarios.

En este caso las vistas son páginas web con extensión GSP, similares a los ficheros JSP de Java, pero con etiquetas posibles por defecto generadas para ser usadas en las GSP, como pueden ser el uso de sentencias condicionales, iteraciones sobre bucles o las etiquetas que sirven para dar formato a las variables mostradas.

Un tipo especial de ficheros GSP con los templates que son partes cuyo contenido se puede incluir en varias páginas GSP, de tal forma que podemos incluir un código común para varias gsp sin tener que repetirlo.

La estructura que sigue un fichero GSP es la de un fichero HTML, con una parte para las cabeceras y otra para el contenido. En las cabeceras podemos definir el título de la página así como si queremos utilizar alguna librería. En el cuerpo podemos encontrar mezclado código HTML junto con etiquetas de la propia GSP. Entre las más destacadas están:

- `g:if`: sentencia condicional que evalúa el contenido del parámetro `test`.
- `g:each`: forma de iterar sobre una lista, un vector o una colección de objetos. Los objetos sobre los que se va a iterar se sitúan en el parámetro `in`, y el objeto actual de la iteración en `var`.

- *g:message*: para mostrar mensajes presentes en los ficheros de internacionalización, que explicaremos más adelante.
- El uso de `${__variable__}`: permite utilizar las variables que envía el controlador como modelo a la vista.

Los ficheros de vistas con los que cuenta la aplicación se estructuran en función del controlador que las utilizan. Presentamos un ejemplo de las Vistas:

```
<%@ page import="es.proyecto.aprenvid.Videos"%>
<!DOCTYPE html>
<html>
<head>
<meta name="layout" content="main">
<g:set var="entityName"
    value="${message(code: 'videos.label', default: 'Videos')}" />
<title><g:message code="default.show.label" args="[entityName]" /></title>
</head>
<body>
<a href="#show-videos" class="skip" tabindex="-1"><g:message
    code="default.link.skip.label" default="Skip to content&hellip;" /></a>
<div class="nav" role="navigation">
    <ul>
        <li><a class="home" href="${createLink(uri: '/home/gestionar')}"><g:message
            code="default.home.label" /></a></li>
        <li><g:link class="list" action="index">
            <g:message code="default.list.label" args="[entityName]" />
            </g:link></li>
        <li><g:link class="create" action="create">
            <g:message code="default.new.label" args="[entityName]" />
            </g:link></li>
    </ul>
</div>
<div id="show-videos" class="content scaffold-show" role="main">
    <h1>
        <g:message code="default.show.label" args="[entityName]" />
    </h1>
    <g:if test="${flash.message}">
        <div class="message" role="status">
            ${flash.message}
        </div>
    </g:if>
</div>
```

```

<ol class="property-list videos">
  <g:if test="${videosInstance?.titulo}">
    <li class="fieldcontain"><span id="titulo-Label"
      class="property-label"><g:message code="videos.titulo.label"
        default="Titulo" /></span> <span class="property-value"
        aria-labelledby="titulo-Label"><g:fieldValue
          bean="${videosInstance}" field="titulo" /></span></li>
    </g:if>
    <g:if test="${videosInstance?.descripcion}">
      <li class="fieldcontain"><span id="descripcion-Label"
        class="property-label"><g:message
          code="videos.descripcion.label" default="Descripcion" /></span> <span
          class="property-value" aria-labelledby="descripcion-Label"><g:fieldValue
            bean="${videosInstance}" field="descripcion" /></span></li>
    </g:if>
    <g:if test="${videosInstance?.asignatura}">
      <li class="fieldcontain"><span id="asignatura-Label"
        class="property-label"><g:message
          code="videos.asignatura.label" default="Asignatura" /></span> <span
          class="property-value" aria-labelledby="asignatura-Label"><g:link
            controller="asignatura" action="show"
            id="${videosInstance?.asignatura?.id}">
              ${videosInstance?.asignatura?.encodeAsHTML()}
            </g:link></span></li>
    </g:if>
  </ol>

```

Figura 78. Vista GSP de Videos - Parte1

```

  <g:if test="${videosInstance?.preguntas}">
    <li class="fieldcontain"><span id="preguntas-label"
      class="property-label"><g:message
        code="videos.preguntas.label" default="Preguntas" /></span>
    <g:each in="${videosInstance.preguntas}" var="p">
      <span class="property-value" aria-labelledby="preguntas-label"><g:link
        controller="preguntas" action="show" id="${p.id}">
          ${p?.encodeAsHTML()}
        </g:link></span>
    </g:each></li>
  </g:if>
  <g:if test="${videosInstance?.subAsignatura}">
    <li class="fieldcontain"><span id="subAsignatura-label"
      class="property-label"><g:message
        code="videos.subAsignatura.label" default="Sub Asignatura" /></span> <span
        class="property-value" aria-labelledby="subAsignatura-label"><g:fieldValue
          bean="${videosInstance}" field="subAsignatura" /></span></li>
  </g:if>
  <g:if test="${videosInstance?.urlVideo}">
    <li class="fieldcontain"><span id="urlVideo-label"
      class="property-label"><g:message
        code="videos.urlVideo.label" default="Url Video" /></span> <span
        class="property-value" aria-labelledby="urlVideo-label"><g:fieldValue
          bean="${videosInstance}" field="urlVideo" /></span></li>
  </g:if>
</ol>
<g:form url="[resource:videosInstance, action:'delete']"
  method="DELETE">
  <fieldset class="buttons">
    <g:link class="edit" action="edit" resource="${videosInstance}">
      <g:message code="default.button.edit.label" default="Edit" />
    </g:link>
  </fieldset>

```

```

<g:actionSubmit class="delete" action="delete"
value="{message(code: 'default.button.delete.label', default: 'Delete')}}"
onclick="return confirm('{message(code: 'default.button.delete.confirm.message', default: 'Are you sure?')}');" />
</fieldset>
</g:form>
</div>
</body>
</html>

```

Figura 79. Vista GSP de Videos - Parte2

La imagen que se observa es la vista que se da al usuario para consultar los datos de un video determinado, en este fichero GSP se van hacer uso de varias etiquetas GSP, a continuación las describimos:

- `<g:set>`: mediante esta etiqueta podemos establecer el valor de una variable de acceso a la página GPS, en este caso se crea una variables *entityName* cuyo valor serán mensajes de error obtenidos por cada unos de las propiedades del video, es decir si existe algún error al momento al mostrar a los usuarios el titulo, la descripción, asignatura, preguntas, urlVideo. Tal y como se observa en las primeras líneas solo está la declaración de la variable con el valor que toma, el uso de la variable aparece en el resto de líneas.
- `<g:message>`: etiqueta cuyo uso es para mostrar los mensajes de error obtenidos por la entidad video, en esta caso se puede observar que el valor obtenido para esta etiqueta mediante el atributo *args* es la variable antes declarada, *entityName*.
- `<g:link>`: esta etiqueta crea un enlace en la vista gsp, gracias a los atributos de los que dispone la etiqueta *controller* y *action* podemos indicar el nombre del controlador y que método de ese controlador se encargará de ejecutar la acción cuando pulsemos sobre el enlace. En las primeras líneas se observa dos etiquetas *g:link* que no disponen del atributo *controller* en este caso no es necesario indicar el nombre del controlador dado que es sobreentendido que debe dirigirse al controlador Video, para justificar el porqué es sobreentendido que se dirigirá al controlador Video es necesario indicar que por cada controlador que se crea en la aplicación se crea una carpeta con su nombre, en este caso disponemos de la carpeta Videos, aquí se agrupan todas las vistas que coinciden con el nombre de los métodos que conforman el controlador Video. Por tanto para los dos enlaces que describimos irán al controlador Video a las acciones *index* y *create*, que una vez realicen las operaciones que se indique en eso métodos nos darán sus respectivas vistas, es decir el primer enlace nos llevan a la lista de todos los videos y el segundo nos lleva al alta de un nuevo video.
- `<g:if>`: el fichero gsp tiene varias etiquetas de este tipo, la tarea de esta etiqueta es crear el contenido con la información que está dentro de la etiqueta siempre que se dé la condición que se especifica en el *g:if*, damos el ejemplo de `<g:if test='${videoInstance.descripcion}'`, esta condición indica si existe un valor para el campo descripción del video, solo en ese caso se mostrará la información, de lo contrario no mostrará nada, este tipo de comprobaciones se hace para cada atributo del Video.

- `<g:form>`: etiqueta usada para formularios, en este caso es necesario usar esta etiqueta para enviar la instancia que deseamos eliminar, cuando pulsemos el botón eliminar. Para el envío de la instancia se usa el atributo *url* aquí se indica en *resource* la instancia a eliminar y en *action* le método que se encargará de ello.
- `<g:actionSubmit>`: etiqueta usada para dirigirnos al método de un determinado controlador, cuando presionemos sobre un botón, para ello disponen de los atributos *controller* y *action*, estos atributos tienen el mismo uso que en la etiqueta `<g:link>`, en este caso al presionar el botón Eliminar aparecerá una ventana emergente preguntándonos si deseamos eliminarlo.

Mediante esta vista de las que dispone nuestra aplicación hemos explicado el uso de algunas de ellas, pero existen muchas más, voy a destacar el uso de dos de ellas que no aparecen en esta vista y son:

- `<g:each>`: etiqueta destinada a recorrer una colección de datos, en nuestra vistas las hemos usado para mostrar la información de las asignaturas y subasignaturas. Se incluye una imagen en la que se observa que con el primer `<g:each>` se recorre la lista de asignaturas y con el segundo `<g:each>` se recorre la lista de las subasignaturas que pertenecen a cada asignatura.

```
<g:each var="asignatura" in="{Asignatura.list().sort{it.nombre}}">
  <h3 class="lista">
    <a style="display: inline;" href="#"> {asignatura?.nombre}
  </a>
</h3>
<ul>
  <g:each var="subAsignatura"
    in="{asignatura?.subAsignatura.sort{it.nombre}}">
    <li>
      <%- {subAsignatura?.nombre}
      --%> <g:link controller="videos"
        action="resultadoBusquedaPrincipal"
        params="{[idSA:subAsignatura?.id]}">
        {subAsignatura?.nombre}
      </g:link>
    </li>
  </g:each>
</ul>
</g:each>
```

Figura 80. Ejemplo de Etiqueta `g:each` usadas en las Vistas GSP

- `<g:render>`: etiqueta usada para incluir información que se repite en las gsp, damos un ejemplo del uso de la etiqueta mediante una imagen. De esta forma solo escribimos el código de la gsp una sola vez y la usamos en cualquier vista que deseamos incluirla mediante `<g:render>`, conviene indicar que la única información que contendrán los *templates* son trozos de código, ya que su información será incluida dentro de una página que ya tiene declarada las etiquetas `<html>`, `<head>` y `<body>`.

```

<div id="showVideo" class="showVideo">
  <g:render template="templates/divShowVideo" />
</div>

<div id="divListTestPreguntas" class="ventanaModal">
  <g:render template="templates/divTestPreguntas" />
</div>

```

Figura 81. Ejemplo de Etiqueta g:render usadas en las Vistas GSP

6.2.4 Servicios

Para no sobrecargar de código a un controlador se hace uso de los servicio, un servicio es una clase que contiene funcionalidad para ser utilizada desde varios controladores y evitar así repetir código. De esta forma, un cambio en la funcionalidad del servicio solo implica el cambio en ese servicio, pero si no existieran, tendríamos código duplicado, lo que haría más costosos los cambios y una mayor dificultad en el mantenimiento de la aplicación.

En nuestro caso vamos a disponer de dos servicios:

- HomeService: en este servicio se han creado dos métodos: *eliminarAcentos* y *emailtoAdmi*. El primero método sirve para remplazar palabras con acentos en el término de búsqueda introducido por el usuario. Se realiza esta acción con el fin de intentar hacer que haya más probabilidad de coincidencias, dado que la consulta se realizaran con el término introducido tanto con acentos como sin ellos. El segundo método sirve para enviar el email al administrador de la aplicación, el email contendrá los datos obtenidos del formulario que ha rellenado el usuario colaborador.
- PreguntaService: este servicio cuenta con tres métodos: *obtenerPrimeraPregunta*, *siguientePregunta* y *anteriorPregunta*. El primero de ellos sirve para obtener el tiempo en el que debe aparecer la primera de las preguntas de un video determinado, el siguiente método sirve para conocer el tiempo en el que debe hacer la siguiente pausa para mostrar la pregunta siguiente del video, este método será invocado una vez se conteste correctamente al actual pregunta y por último el tercer método indica el tiempo al que debe retroceder el video en el caso que se haya contestado de forma equivocada a una pregunta.

Cabe destacar que la aplicación usas servicios de dos plugins instalados, uno de ellos es el servicio *mailService* usado desde nuestra servicio HomeService para el envío del email y el otro servicio es *simpleCaptchaService* usado para comprobar si los caracteres introducidos de la imagen captcha que aparece en el formulario de colaboración son correctos. La Figura 82 muestra el ejemplo del servicio HomeService.


```

class HomeService {

  def mailService

  String eliminarAcentos(String input){
    String original = "áàäëèëíîïóòóúûüñÁÀÄÊËËÎÏÓÒÒÚÛÜŃÇç"
    String ascii = "aaaaeeiiiiooouuunAAEEEEEIIIOOOUUUNcç"
    String output = input
    for (int i=0; i<original.length(); i++) {
      output = output.replace(original.charAt(i), ascii.charAt(i))
    }
    return output;
  }

  def emailtoAdmin(String nombre, String apellido, String email,
    String profesion, String sexo, String comentario){

    mailService.sendMail {
      //to "ssaniluap@gmail.com"
      to "aprenvideos@gmail.com"
      //from "paulinalex676@hotmail.com"
      from "aprenvideos@gmail.com"
      subject "Colaborar con Aprenvid"
      body(view: "/home/mensajeEmail",
        model:[nombre:nombre,apellido:apellido,email:email,
          profesion:profesion,comentario:comentario])
    }
  }
}

```

Figura 82. Servicio HomeService

Se observan los dos métodos anteriormente mencionados y como se hace uso del servicio mailService, para el empleo de este servicio es necesario haber instalado el plugins Mail previamente. Este plugins es útil a la hora del envío de correos, se envía el email indicando la dirección origen, la dirección de destino, un asunto y el cuerpo del mensaje, en este caso el cuerpo es una vista a la que pasamos varios argumentos, en función del contenido del mensaje.

6.2.5 JavaScript

El uso de javascript en la aplicación tiene dos misiones importantes, una de ellas es mejorar el aspecto visual de la aplicación y la otra es permitir interactuar a los usuarios con los videos mediante las preguntas que aparecen a lo largo de la reproducción.

La funcionas que realizan los ficheros .js son las siguientes:

- Mostrar ventanas *dialog*: son ventanas emergentes que sombrean el contenido de la página original para centrarse en la pequeña ventana que acaba de aparecer, se usa este efecto para mostrar las preguntas de los videos.
- Utilizamos *datatable*: nos permite darle dinamismo a nuestras tablas, mejorando su aspecto, permitiendo paginar resultados, hacer búsquedas y ordenar por campos la información.

- Efecto *accordion*: permite mostrar diversos contenidos clasificados en secciones de manera que se mantenga abierta una de esas secciones y se cierre automáticamente al abrir otra.
- Otra función importante son las llamadas *.ajax* que permite realizar acciones en una página web que necesitan respuestas de un servidor sin recargarla. Con esto se consigue que la web sea dinámica y por tanto más atractiva.

En la imagen se presenta un ejemplo de uso de código JavaScript en la aplicación:

```
function ajaxComprobarSecuenciaVideo(idVideo,idPregunta2,ok){
    var tiempoMostrarPregunta, tiempoInicio;
    $.ajax({
        url: "/AprenVid/videos/comprobarSecuenciaVideo",
        type: "POST",
        data: "iddV="+idVideo+"&iddP="+idPregunta2+"&ok="+ok,
        success: function(html) {
            $("#divShowVideo").html(html);

            if($('#accion').attr('value')=="avanzar"){
                controlaMomentoParaMostrarPreguntas();
            } else if($('#accion').attr('value')=="retroceder"){
                controlaMomentoParaMostrarPreguntas();
            }
        }
    });
};
```

Figura 83. Uso de JavaScript para Videos

Se puede observar el uso de la función *.ajax* que está dentro de la función *ajaxComprobarSecuenciaVideo*, el objetivo de esta función es llamar al método *comprobarSecuenciaVideo* del controlador *Video* pasándole como parámetros el identificador del video, la pregunta y la respuesta marcada por el usuario, acto seguido el método del controlador realiza las operaciones necesarias de esta manera al volver al video para su reproducción sabremos si continua desde el punto que se interrumpió para mostrar la pregunta o por el contrario debe repetir el último tramo del video antes de mostrar la pregunta.

6.2.6 Conf

Los ficheros de configuración que tiene Grails y las funciones de cada uno de ellos, son las siguientes:

- *BootStrap*: este fichero contiene las partes de código que se deben ejecutar al arrancar la aplicación y al destruirla.
- *BuildConfig*: en este fichero podemos encontrar la configuración sobre la construcción del proyecto en un fichero WAR, los repositorios para que descargue las dependencias o los *plugins* que utiliza.
- *Config*: desde este fichero se puede configurar los distintos aspectos de la aplicación, como de cada uno de sus *plugins*. Por ejemplo, en este fichero se configura, en función del entorno en el que se encuentre desplegada la aplicación, si se envían correos electrónicos o no.

- *DataSource*: en este fichero se definen las bases de datos a utilizar y la política (si se crea una base de datos nueva cada vez que arranca o si se actualiza la que ya existiera), en función del entorno en el que nos encontremos.
- *UrlMappings*: define los controladores a los que se redirigir en caso de error, cuando se accede a la URL de acceso, o cómo se muestra en la URL cuando se interactúa con controladores y sus métodos.

CAPÍTULO 7

7. Pruebas

En este capítulo se detalla las diferentes pruebas realizadas a la aplicación para comprobar que las funcionalidades implementadas funcionan correctamente.

7.1 Pruebas realizadas

La etapa de pruebas es llevada a cabo según se crea la funcionalidad de la aplicación ya que es necesario verificar que la tarea satisface todos los requisitos.

7.1.1 Pruebas de Caja Blanca

Para este tipo de pruebas se introducen datos desde la vista de la aplicación y se observa el comportamiento de la misma.

Para cada vista se han realizado una serie de pruebas que se detallan a continuación:

7.1.1.1 Pruebas de Búsqueda General



Figura 84. Prueba Unitaria sobre Búsqueda General

Para la vista mostrada en la Figura 84 se han llevado a cabo las siguientes comprobaciones:

- No introducir ninguna palabra clave de búsqueda y dar al botón buscar, en este caso aparecerá un mensaje indicando que debe rellenar el campo solicitado, este mensaje aparecerá como una ventana emergente sobre el campo donde debemos introducir el término de búsqueda.
- Se introduce cualquier tipo de valor para el término de búsqueda, combinación de caracteres distintos al alfabeto, signos de puntuación, numéricos, etc. Todos estos son aceptados.
- Introducir una palabra clave de búsqueda, y observar los resultados obtenidos.
- En el menú de las asignaturas se ha probado a pinchar en la subasignatura que deseamos obtener una lista de resultados y comprobamos que los resultados pertenecen efectivamente a las subasignatura que hemos seleccionado.

The screenshot shows the AprenVid website interface. At the top, there's a green header with the 'AprenVid' logo and the title 'Aprendizaje Mediante Videos'. A 'Colaborar' button is in the top right. On the left, a sidebar lists various subjects: Arte, Cine, Historia del Arte, Música, Astronomía, Biología, Diseño de Bases de Datos, Economía, Filosofía, Física, Geografía, Historia, Idiomas, Inteligencia Artificial, and Matemáticas. The main content area is titled 'Resultados obtenidos con el término de búsqueda "Matemáticas"'. It features a search bar with 'Show 25 entries' and a search input field. Below this is a table with two columns: 'TITULO' and 'DESCRIPCION'. The table lists several results related to mathematics, including 'Definición de Sucesiones Matemáticas', 'Demostración de la Fórmula de Esperanza Matemática', 'Eduardo Sáenz de Cabezón: Las matemáticas son para siempre', and 'Ingeniería matemática'.

TITULO	DESCRIPCION
Definición de Sucesiones Matemáticas	Definición de Sucesiones Matemáticas Veremos qué es una sucesión de números reales, y algunas formas de describirlas. Luego, estudiaremos algunas de sus propiedades, y las clasificaremos en monótonas y no monótonas.
Demostración de la Fórmula de Esperanza Matemática	Veremos la fórmula general que nos servirá para calcular la esperanza matemática para variables aleatorias. Demostraremos la misma de una forma simple y sencilla.
Eduardo Sáenz de Cabezón: Las matemáticas son para siempre	¿Cómo usar las matemáticas para expresar tu amor por otra persona? Eduardo Sáenz de Cabezón nos da una respuesta muy inesperada.
Ingeniería matemática	Existen algunas carreras de las que se tiene una idea clara sobre cuáles son las materias que se cursan, pero no es del todo claro cuál es

Figura 85. Resultado de Prueba sobre Búsqueda General o Por SubAsignaturas

Las pruebas realizadas para obtener esta imagen han sido:

- Escribir el término de búsqueda "Matemáticas " y Verificar los resultados.
- En el caso de no disponer información con el término de búsqueda escrito por el usuario se obtendrá un mensaje indicando que no se han encontrado resultados, este mismo mensaje aparece en el caso que no haya una lista de resultados para la subasignatura seleccionada.

7.1.1.2 Pruebas con Formulario de Colaboración

The screenshot shows the 'FORMULARIO DE COLABORACION' (Collaboration Form) on the AprenVid website. At the top, it says 'Si desear que colguemos tus videos para ayudar AprenVid en la labor de Aprendizaje, por favor rellena este formulario y nos pondremos en contacto contigo.' The form includes fields for 'Nombre', 'Apellido', 'Email', 'Profesion', 'Sexo' (with radio buttons for 'Masculino' and 'Femenino'), and a large text area for 'Comentarios'. Below the comments field, there's a CAPTCHA challenge that says 'Escriba los caracteres que ve por favor' followed by the image 'QKLBQT'. At the bottom, there are two buttons: 'Enviar Datos' and 'Cancelar'.

Figura 86. Prueba Unitaria sobre Formulario de Colaboración

Pruebas realizadas para el Formulario de Colaboración:

- Rellenar el formulario con todos los datos correctos y pulsar el botón enviar, el resultado será que los administradores reciben un correo con estos datos.
- Tratar de enviar el formulario sin alguno de los datos requeridos, como resultado obtenemos un mensaje indicándonos que rellenemos el dato que falta, además de resaltar el campo.
- Enviar el formulario con el campo de email no válido, es decir que no siga el patrón de correo electrónico, aparecerá un mensaje indicándonos que el email no es válido.
- Si el campo reservado para introducir los caracteres de la imagen captcha es rellenado por caracteres que no aparecen en la imagen cuando pinchamos en el botón de enviar aparecerá un mensaje indicándonos este hecho.

7.1.1.3 Pruebas al interactuar con videos

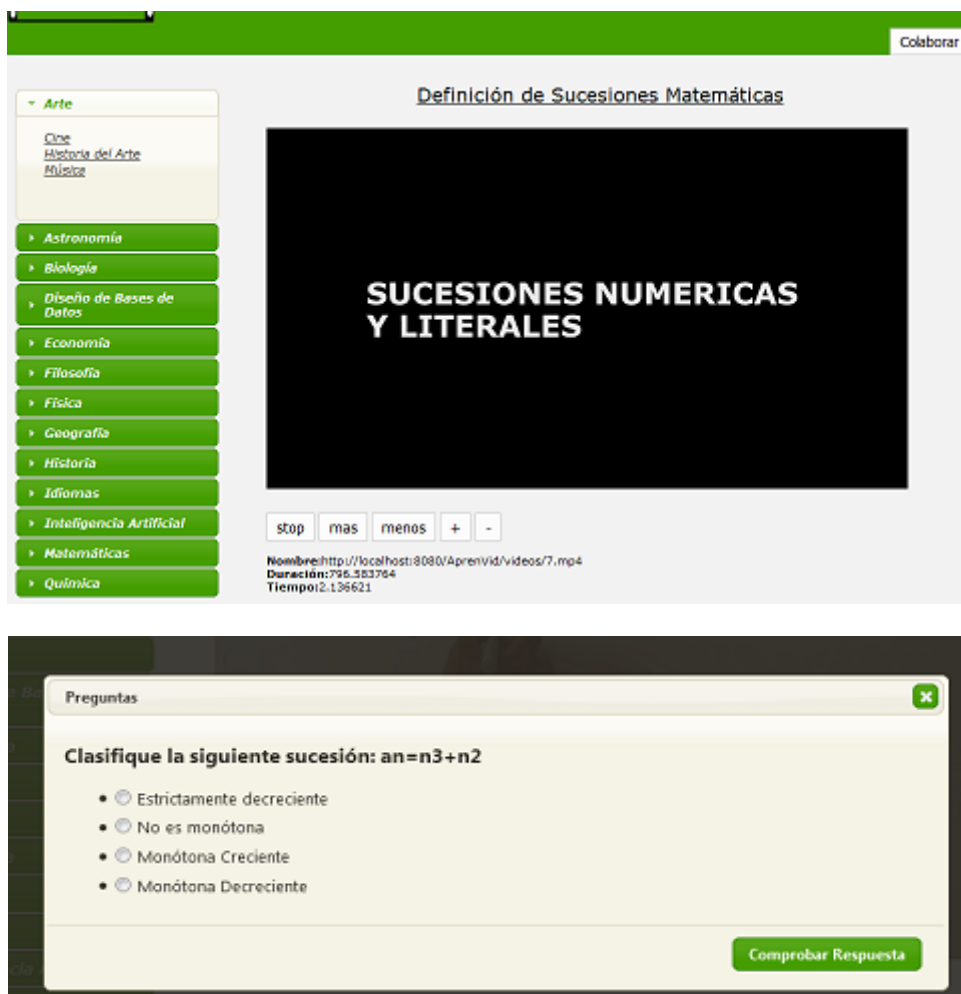


Figura 87. Prueba Unitaria en Reproducción de Videos

En las imágenes anteriores se observa la reproducción de un video seleccionado y la pregunta que se ha establecido que aparezca en ese momento determinado, las comprobaciones que se realizan son:

- Si el video no está disponible al presionar el botón Play, la aplicación no realiza ninguna acción.
- Cuando un video está disponible al presionar Play el video empieza con la reproducción, en los momentos establecido aparece la pregunta en una ventana modal tal y como muestra la Figura 87, al ser preguntas tipo test se da una lista de respuestas, de entre las cuales, es necesario seleccionar alguna de ellas, si no seleccionamos ninguna respuesta aparecerá un mensaje indicando que "Debemos seleccionar una respuesta".
- Al visualizar las preguntas y elegir alguna de las respuestas se ha comprobado que el sistema indica mediante un mensaje si la respuesta proporcionada ha sido correcta o incorrecta.
- Una vez nos indique el sistema si la respuesta es correcta el siguiente paso es revisar si la reproducción del video actúa según las respuesta proporcionada, es decir si continua la secuencia normal al contestar una respuesta correcta o si retrocede al contestar con una respuesta incorrecta.
- Se comprueba además que aparecen todas las preguntas en los tiempos establecidos y que al llegar al final de la reproducción, si deseamos volver a reproducirlo este tendrá el mismo comportamiento, es decir reproducirá el video desde el principio y mostrará todas las preguntas ya vistas en la primera reproducción y en los mismos momentos.

7.1.1.4 Pruebas con Administrador

Figura 88. Prueba Unitaria en Página de Administrador

En la imagen se puede observar la ventana principal para el administrador, en esta vista se realizan las siguientes acciones:

- En primer lugar es necesario rellenar los campos solicitados, en caso de no hacerlo aparecen un mensaje indicando que debemos rellenar el campo que no lo hayamos hecho.
- Una vez escrito el usuario y la contraseña se comprueba si los datos introducidos son correctos, en el caso de ser así permite el acceso.
- La imagen que se observa a continuación es la que ve el administrador siempre que se le haya permitido el acceso.



Figura 89. Resultado de Prueba Unitaria en Página del Administrador

7.1.1.5 Pruebas Gestionar

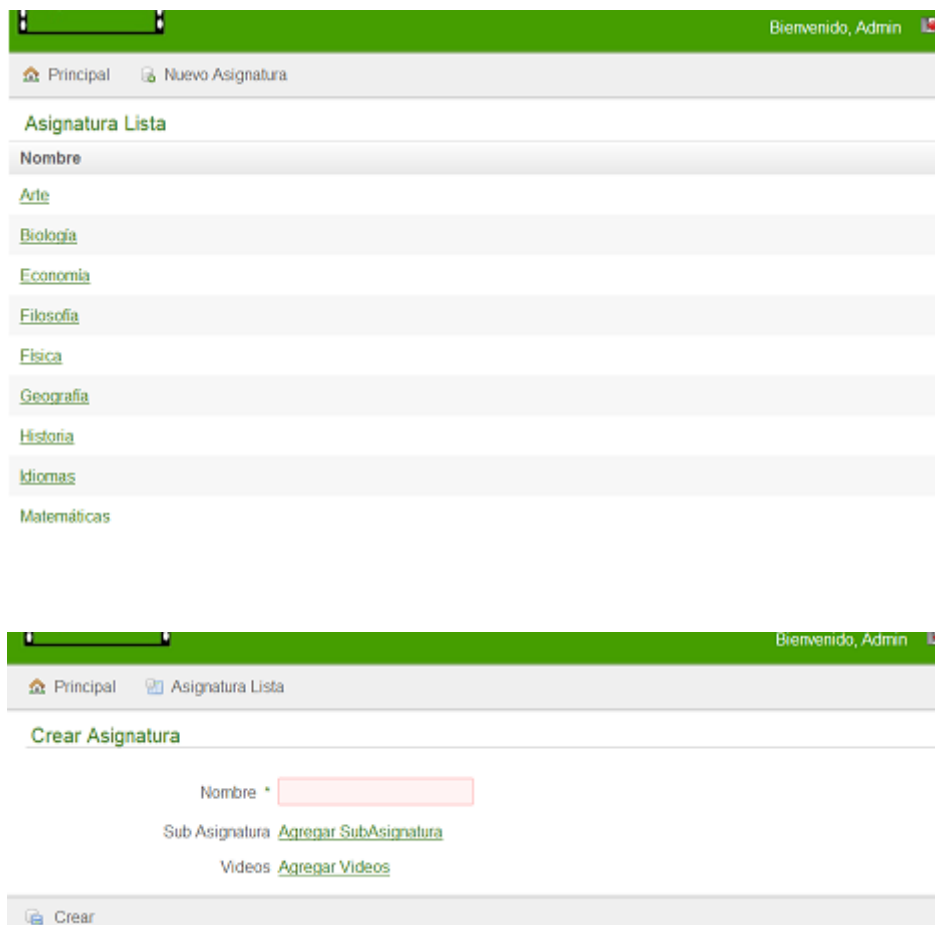


Figura 90. Prueba Unitaria en Gestionar Asignaturas

- Pruebas para el Alta

En las anteriores imágenes se observa la ventana principal para Asignaturas y la ventana que permite dar de alta una nueva asignatura, el tipo de comprobaciones que se detallan a continuación son perfectamente válidas para asignaturas, subasignaturas, videos y preguntas.

- Introducir los datos correctamente.
- Introducir los datos sin algún dato no requerido.

- Introducir datos dejando en blanco alguno de los requeridos, esto provoca un error, por lo tanto no crea la instancia solicitada en la base de datos.
- Para el campo nombre de asignaturas y subasignaturas debe ser un valor único, si pretendemos dar de alta una asignatura o subasignatura ya existente en la base de datos obtenemos un mensaje indicándonos que introduzcamos un valor único.
- Otro dato a tener en cuenta para proceder con el alta de una asignatura, subasignaturas o videos, es necesario introducir una nombre para asignaturas y subasignaturas como mínimo de tres caracteres y máximo 50, para el campo título de video, también se ha establecido la misma longitud, si la longitud del nombre de asignatura o subasignatura o para el título de video no están dentro de este rango no permite el alta y lanza un mensaje de aviso.
- Para proceder con el alta de una pregunta es necesario que los campos cumplan un rango establecido, para el enunciado y respuestas se establece una longitud de 3...100 caracteres, para el campo correcto se pueden elegir valores del 1...4, ya que se tiene que corresponder con alguna de las respuestas, es decir 1 si la respuesta correcta es la respuesta1, 2 si la respuesta correcta es respuesta2, etc, para ordenpregunta se establece un rango de 1...60, minutos que contiene una hora. En todos estos casos hemos probado a introducir valores por debajo y por encima de las longitudes establecidas y se verifica que dan los correspondientes errores sin procede al alta.
- En cada unos de los campos tipo texto se han probado varios datos, como caracteres diferentes al alfabeto, signos de puntuación, números, etc. Todos ellos son aceptados.

- **Pruebas para el Borrado**

Pruebas realizadas al intentar la gestión de borrado, tanto para Asignaturas, SubAsignaturas, Videos y Preguntas.

- Se prueba a borrar asignaturas, subasignaturas, videos y preguntas registrados en la aplicación, este borrado se realizará de manera exitosa siempre y cuando no haya dependencias para asignaturas, subasignaturas y videos. Las preguntas serán borradas sin ningún problema.
- Se prueba a borrar asignaturas, subasignaturas y videos que tiene dependencia entre ellas y se comprueba que proporciona un error de clave foránea impidiendo borrar las instancias de la base de datos.

- **Pruebas para la Modificación**

- Al realizar modificaciones tanto en asignaturas, subasignaturas, videos y preguntas se comprueban las mismas operaciones como que de un alta se tratará, es decir que al editar algunas de estas entidades deben seguir cumpliendo con las condiciones de longitud, valores únicos y rangos numéricos fijados.

- **Pruebas para el Listado**

- El listado no requiere realizar ningún tipo de prueba dado que con solo pinchar sobre el botón Gestionar Asignatura, Gestionar SubAsignatura, Gestionar Videos o Gestionar Preguntas obtendríamos el directamente el listado correspondiente, sin necesidad de rellenar ningún dato.

CAPÍTULO 8

8. Conclusiones y Trabajos Futuros

Este capítulo enumera las conclusiones alcanzadas tras el desarrollo de la aplicación y las principales líneas de trabajo futuro.

8.1 Conclusiones del proyecto

En este apartado incluiremos las conclusiones obtenidas en los siguientes puntos:

- Uso de tecnologías ágiles
- Pruebas realizadas
- Conclusión personal del proyecto

8.1.1 Uso de tecnologías ágiles

Para llevar a cabo este proyecto ha sido necesario un estudio de las tecnologías actuales con el fin de controlar de forma automático el objeto principal de esta aplicación, los videos de aprendizaje.

8.1.1.1 Html5

Hasta hace relativamente poco, los desarrolladores web que querían incluir elementos dinámicos y animaciones en su web prácticamente solo podían utilizar Flash, creada originalmente con Macromedia en los noventas y adquirida por Adobe en 2005, Flash revolucionó el mundo web y se convirtió rápidamente en el estándar multimedia de internet.

Así ha sido hasta el despegue definitivo en los últimos meses de HTML5, que ya se considera un sustituto natural y el nuevo estándar para multiplicar la interactividad de los proyectos web.

Podemos nombrar algunas ventajas que ayudan a un desarrollador a utilizar HTML5

- Posicionamiento en buscadores: el contenido en HTML es leído e indexado muy fácilmente por lo buscadores, acción que con la tecnología Flash no lo hace.
- Dispositivos móviles: sin duda el auge de la movilidad, de tablets y smartphones, ha sido determinante para que HTML5 avance posiciones respecto a Flash.
- Potencia gráfica y animación: con HTML5 podemos realizar la mayoría de animaciones que antes podíamos hacer con Flash.
- Accesibilidad: muchas personas con deficiencias visuales acceden al contenido de Internet gracias a aplicaciones de síntesis de voz que leen el contenido que reflejan las pantallas de sus navegadores. Estos sistemas han perfeccionado la "lectura" del contenido en HTML y de las etiquetas que jerarquizan este lenguaje. Existen más dificultades con las películas resultantes de Flash.
- Coste: HTML5 y todos sus elementos componen un lenguaje libre de uso, que no requieren ningún coste adicional para los desarrolladores, Flash es un programa para el que es necesario adquirir una licencia.

8.1.1.2 Javascript

Es un lenguaje referente en cuanto a lenguaje de programación web, sus utilidades para retoques visuales y las interfaces de usuario. Gracias a su extensibilidad a través de librerías podemos utilizar múltiples elementos, como menús, pestañas, tablas dinámicas, etc.

Gracias a JQuery el uso de Javascript es más sencillo por la cantidad de funciones que implementa esta librería, convirtiéndose sin duda en la más importante del lenguaje.

JQuery-ui otra de las librerías destacadas que permite gran cantidad de operaciones principalmente ofrece librerías de interfaces de usuarios con colores personalizables, ampliando así las posibilidades de JQuery.

En este proyecto hemos hecho uso de este tipo de tecnologías que antes eran desconocidas permitiendo así asentar conocimientos ya conocidos y aprendizaje de nuevos conceptos y mecanismos que han permitido aportar detalle visual en la aplicación.

8.1.1.3 Groovy y Grails

El uso de un lenguaje como Groovy y el framework de desarrollo proporcionan sencillez en su sintaxis junto con una gran librería de plugins y extensiones que se pueden utilizar en los proyectos.

Con el uso de Groovy se puede echar mano de closures (segmento de códigos asignados a una variable que se ejecutan cuando deseemos), el tipado débil de variables, etcétera.

Grails es una gran ayuda en el mapeo relacional con las bases de datos, lo que permite utilizar consultas dinámicas, programar las vistas de una forma sencilla mediante etiquetas que vienen por defecto, pasar a la vista datos desde el controlador de manera asequible.

La unión de Groovy y Grails ofrece un conjunto de herramientas que encajan a la perfección y que ofrecen una característica muy destacada, como es la posibilidad de realizar cambios en tiempo de ejecución, lo que ahorra tiempo durante el desarrollo y permite solucionar errores de una manera más rápida.

Desde luego la unión de Grails and Groovy son una herramienta que en su conjunto ayudan a desarrollar una aplicación desde cero de una manera más ágil de lo que pueden ofrecer otro tipo de framework como por ejemplo J2EE que abordan framework más complejos como struct2, spring o hibernate, que requieren configuraciones previas algo complejas para llevar a cabo una aplicación desde cero.

A pesar de las ventajas que ofrece Grails and Groovy también caben destacar los inconvenientes que puede proporcionar en algún punto del desarrollo del proyecto, es un framework que se ha iniciado desde julio de 2005 y hasta el momento no tiene el auge que tiene Java(con más años de vida en el mercado), siendo este el inconveniente que indicamos ya que a la hora de proporcionar un error en la aplicación hay un serie de ellos de los cuales no es posible encontrar ayudas en internet, por lo que la solución de esos errores pueden ser en algunos casos una misión imposible, por lo tanto podemos resaltar la falta de documentación sobre todo cuando no compartes conocimientos en este lenguaje con alguna persona con más tiempo familiarizadas con él. Otro punto a tener en cuenta a la hora de utilizarlo son las continuas nuevas versiones que ofrecen de Grails en las que incluyen modificaciones bastante importantes, eliminando en muchos casos clases deprecated (clases en desuso) que hacen inutilizable plugins importantes, siendo usados en versiones anteriores de una manera eficaz y sencilla.

8.1.2 Pruebas

La etapa de pruebas es un capítulo importante realizado en cualquier tipo de proyecto, ya que permiten conocer si las tareas han sido desempeñadas correctamente y permite la búsqueda de posibles errores que no han sido considerados sobretodo en la ciencia de la informática, tan propensa a fallos.

Se han realizado las pruebas unitarias necesarias para verificar el funcionamiento de todos los métodos que contiene nuestra aplicación, estas pruebas son sencillas de implementar y ver donde se producen los errores depurando el código hasta encontrar el fallo y solventarlo de la mejor manera posible validando así la marcha correcta del método a comprobar.

Es por tanto importante que las pruebas realizadas cubran todos los casos posibles, para tener una seguridad de que la aplicación cumple con los objetivos de funcionalidad y evitar así futuros errores que conlleven mayor gasto en tiempo y coste para el proyecto, aunque como ya destacamos antes la informática es una ciencia propensa a errores, y podemos encontrarnos escenarios que conlleven a errores que se nos han pasado por alto por algún motivo, una vez hallado este escenario, debemos solucionarlo lo más pronto posible para que el efecto sea mínimo.

8.1.3 Conclusión Personal

Llevar a cabo este proyecto ha hecho que tenga conocimientos sobre tecnologías poco usadas por mi parte y permitan marcarme retos personales para afianzar más aún los conocimientos adquiridos, además del continuo aprendizaje de otro tipo de tecnologías usadas actualmente permitiendo así aumentar mi formación ya que hoy en día es muy importante estar al día puesto que continuamente la informática avanza a pasos de gigante.

Dar la oportunidad de interactuar con los videos de aprendizaje, lo veo como una herramienta didáctica importante que en el futuro pueden llegar a ser de gran utilidad para las personas que desean aprender ya que mediante las preguntas que nos hace el programa sobre el video que estamos reproduciendo nos permiten valorar en tiempo real si este video nos ayuda a aprender el tema tratado, cabe destacar también que las preguntas pueden ser un método de autoevaluación para el usuario. Cuántas veces hemos reproducido un video y nosotros mismos hemos tenido que pausarlo para realizar el mismo ejercicio que se está realizando en el video y obtener una respuesta que será comparada con la que se proporciona en el video una vez iniciemos nuevamente la reproducción. Por este motivo en lugar de hacernos nosotros las preguntas como oyentes que mejor opción que sea la aplicación quien haga las preguntas y nosotros tan solo contestarlas.

Resalto la utilidad de los videos hoy en día como herramienta de aprendizaje ya que predominan ante los textos, que se pueden considerar aburridos dado que los videos ofrecen imagen y voz que cautivan más al internauta obteniendo resultados más rápidos que los adquiridos al leer el contenido de una página web sobre el tema consultado. Yo misma para el desarrollo de este proyecto he tenido que ver videotutoriales y aprender de ellos para adquirir y entender los conceptos necesarios, no digo que no haya tenido que consultar texto en muchas páginas pero si digo que mediante los videos ha sido más fácil realizar configuraciones

de cualquier tipo ya que vas realizando los pasos poco a poco pausando el video en los momentos necesarios.

8.2 Trabajos Futuros

Dentro de los trabajos a tener en cuenta para el futuro destacaremos los siguientes:

- Mejoras en las **"Bases de Datos"**: dotar a la aplicación de un mayor número de videos para cada una de las asignaturas registradas, con esta mejora podemos hacernos con una amplia base de datos, aspecto importante para que el usuario final pueda hacer uso de la aplicación. Para este punto es imprescindible disponer con la colaboración de profesores, estudiantes que impartan clases en las aulas, o cualquier persona que tenga la capacidad de enseñar.
- Ampliar el **"Catálogo de Búsqueda"**: es posible pensar en ampliar el catálogo de búsqueda, atendiendo no solo a las asignaturas y subasignaturas, si no a los temas y subtemas que se imparten en cada una de ellas, se puede incluso extender a la clasificación de cada tema y subtema por nivel de enseñanza, es decir, primaria, secundaria o universitaria.
- Mejora en la **"Reproducción Automática de los Videos"**: actualmente el sistema esta funcionando de tal manera que al volver al video, una vez respondida las preguntas tipo test que aparecen automáticamente, es necesario pinchar en el botón PLAY tanto como para continuar con la reproducción del video o repetir algún tramo, claro está que estas acciones dependen del tipo de respuesta proporcionada por el usuario, sería más fácil para usuario que el video se reproduzca automáticamente una vez haya contestado a la pregunta, bien sea para continuar o para repetir un tramo del video.
- Controlar el **"Tiempo de duración de los Videos"**: de momento la aplicación no dispone de ningún tipo de control en cuanto al tiempo de duración de los videos, con este tipo de control solo permitiríamos reproducir videos que por ejemplo no superen los 15-20 minutos de duración, pensar en videos que duren 60 minutos no sería factible por dos razones.
 - La interacción con el video no se realizaría de manera fluida, tener que espera a que aparezcan las preguntas automáticamente se podría convertir en una tarea interminable llegando aburrir al usuario, y sin pensar en que conteste de forma equivocada, tendría que reproducir un tramo muy extenso del video para que la pregunta aparezca nuevamente. Ofrecer videos con tiempos de duración extensos perjudica la interacción del usuario.
 - Los videos ocupan un espacio enorme, por este motivo es necesario albergar videos de corta duración para que nuestra colección de videos sea lo más amplia posible y no tener problemas de espacio con pocos videos de larga duración.
- Controlar el **"Tiempo en el que aparezcan las preguntas"**: establecer un control en cuanto a los minutos en los que deben aparecer las preguntas. A continuación detallamos dos tipos de control que se deberían realizar:

- No permitir introducir en el campo OrdenPregunta un tiempo superior a la duración del video, esto puede ocasionar que la pregunta no aparezcan nunca a lo largo del video.
- Impedir que se establezca el mismo tiempo "OrdenPregunta" para distintas preguntas asociadas al mismo video, esto podría llevarnos a un bucle.

De momento no supone inconvenientes ya el administrador de las altas de las preguntas y lo controla directamente él.

- Permitir **"Registrarse a los Usuario Colaboradores"**: en el caso de que haya algún usuario que desee colaborar con la aplicación cediéndonos sus videos de aprendizaje actualmente disponemos de una formulario en el que los usuarios colaboradores nos pueden hacer llegar sus datos de contacto, pero pensando en el futuro podríamos crear un módulo de registro en la aplicación con el fin de permitirle subir videos a un repositorio que almacene una carpeta con un identificar único y sea exclusivo para su uso. Los videos de estas carpetas y la información que nos hagan llegar los usuarios colaboradores deben ser revisadas por las personas cualificadas y los administradores de la aplicación para deliberar si los videos son apropiados para formar parte de nuestro catálogo.

La tarea de subir videos a la aplicación mediante un registro del usuario acarrea otro tipo de labores que son:

- Formulario de Alta de Usuario Colaborador: el primer paso es permitir acceder al usuario colaborador, para ello necesitamos un formulario mediante el cual se pueda proceder al alta en la aplicación y se le permita el acceso con un perfil algo parecido al del administrador, pero bastante limitado. Solo podrá subir los videos y subir las preguntas asociadas a ese video. Será necesario crear tablas distintas a la que usa el administrador para el registro tanto de los videos como de las preguntas. Además será necesario permitir al usuario colaborador "Gestionar Videos" y "Gestionar Preguntas", ya que debemos ofrecerle la posibilidad de crear, editar, modificar, eliminar y listar los videos y las preguntas.
- Formularios de Alta de Videos: crear vistas de Alta de Videos que serán gestionadas por los usuarios que se registren en la aplicación.
- Formulario de Alta de Preguntas: una vez haya rellenado el formulario de Alta de Video es necesario rellene el formulario de preguntas que acompañaran a los videos, por lo tanto se debe crear una vista de Alta de Preguntas que será gestionado por el usuario colaborador.
- Tanto para el Alta de Videos como para el Alta de Preguntas es imprescindible tener ya el control "tiempo de duración de los videos" y el "tiempo en el que aparezcan las preguntas".
- Establecer un bloque de Usuarios Malintencionado: es decir si los videos que nos están haciendo llegar son videos de carácter no educativo hay que impedirles que sigan subiendo contenido al sistema puesto que están ocupando un espacio valioso con contenido inútil.
- Aviso al usuario Colaborador: es necesario avisar al usuario colaborador si los videos que nos han facilitado han sido subidos en la aplicación de forma exitosa o por el contrario es necesario que el usuario colaborador rectifique algún dato en la información que nos han hecho llegar o si es necesario que

cambie algo en el video, previamente claro está, haya sido valorado por una persona cualificada.

No se ha pensado como trabajo futuro permitir a un usuario colaborador subir directamente los videos a la aplicación y los vea colgados en ese mismo momento, ya que supone un riesgo para la aplicación dado que no disponemos de ningún tipo de seguridad que nos garantice a los administradores de la aplicación que subirán videos de carácter puramente educativo. Este punto puede llegar a ser un poco más difícil de implantar incluso a futuro ya que los videos aun siendo de carácter educativo es necesario que sean revisados por personas cualificados que valoren la calidad de los mismos.

- Dar otro tipo de posibilidades al **"Interactuar con los Videos"**: se puede pensar en permitir al usuario contestar las preguntas que aparecen automáticamente sin que estas condicionen la reproducción normal del video, si no que una vez haya terminado el video, se de la valoración a las preguntas contestadas, es decir el número de preguntas contestadas de forma correcta e incorrecta, y mediante estos resultados obtenidos sea el programa quien decida si es necesario reproducir un video de nivel inferior, superior o incluso repetir el mismo video.
- Crear una aplicación **"Descargable para Android"**: dar la posibilidad a los usuarios que desean aprender asignaturas como una aplicación descargable en su teléfono móvil, sería de mayor utilidad y facilitaría su uso, ya que aprovecharían cualquier momento del día para emplearla y no tendrían que esperar a llegar a casa para usar el ordenador. Para llevar a cabo esta tarea es necesario tener en cuenta dos aspectos:
 - Crear un Webservice: mediante la creación de un web Service o Servicio Web se pueden intercambiar datos entre aplicaciones distintas, podríamos pensar en usar el estándar REST, que haciendo uso del protocolo HTTP, proporciona un API que utilizan cada uno de los métodos (GET, POST, PUT, DELETE, etc) para poder realizar diferentes operaciones entre la aplicación que ofrece el servicio web y el cliente. Para facilitarnos un poco la tarea por parte de GRAILS existe el plugins REST que podríamos utilizar para este fin.
 - Aprender Android: es imprescindible entre otras cosas, conocer como Android, que en este caso sería el cliente, consumiría el Webservice y no solo bastaría con eso, si no que deberíamos estudiar de lleno y crear lo que deseamos, una aplicación descargable.

Crear una aplicación descargable para Android no quiere decir que no puedan ver la aplicación en el navegador de su dispositivo móvil ya que gracias a HTML5 esta tarea es posible.

- Dar la oportunidad al usuario de indicarnos su **"Calidad de Experiencia"**: permitir de alguna forma al usuario transmitirnos su placer o disgusto de la aplicación o de algún video, para conseguir esto, de momento se ha pensado en crear un foro de opiniones.
- Disponer de **"Personas Cualificadas"**: es necesario disponer de un docente que desee cooperar con la aplicación con el fin de evaluar los aspectos pedagógicos , aspectos funcionales, aspectos técnicos y expresivos de los videos didácticos, valorar estos aspectos con un personal cualificado ayudaría a que la aplicación ofrezca una utilidad seria y eficaz para los alumnos que confían en el proyecto.

CAPÍTULO 9

9. Biografía y Referencias

- [1] Pere Marqués, "**¿Por qué las TIC en Educación?**", Madrid Octubre 2012.
<http://peremarques.blogspot.com.es/2012/10/por-que-las-tic-en-educacion-que.html>
- [2] Juan Luis Bravo Ramos, "**Los Medios Didácticos en la Enseñanza Universitaria**", Madrid, Febrero 1998, pp 5-7. <http://www.ice.upm.es/wps/jlbr/Documentacion/Libros/tecnorec.pdf>
- [3] Youtube en la Enseñanza, "**Videos Educativos**", Publicado por GPD.
<http://gpd-ada.blogspot.com.es/2008/01/vdeos-educativos.html>
- [4] Milos Ljubojevic, Vojkan Vaskovic, Srecko Stankovic, Jelena Vaskovic, "**Revista Mexicana de Bachillerato a Distancia**", Febrero 2015.
<http://bdistancia.ecoesad.org.mx/?articulo=el-uso-de-video-complementario-en-la-ensenanza-multimedia-como-herramienta-didactica-para-incrementar-la-eficiencia-del-aprendizaje-y-la-calidad-de-experiencia>
- [5] Cristina Elizabeth Guevara, "**Herramientas Didácticas como Apoyo a la Educación**", Noviembre 2013, pp 1-11.
http://es.slideshare.net/cristina_guevara/herramientas-didacticas-como-apoyo-ala-educacion
- [6] Gerardo Chunga, "**Herramientas libres para crear actividades educativas multimedia**".
<http://www.profesoronline.net/2014/07/15/conoce-10-herramientas-libres-para-crear-actividades-educativas-multimedia/>
- [7] "**Entendiendo HTML5: Guía para principiantes**", Mayo 2013.
<http://hipertextual.com/archivo/2013/05/entendiendo-html5-guia-para-principiantes/>
- [8] Enrique González, "**¿Cuales con las versiones de HTML?**".
http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=444:icuales-son-las-versiones-de-html-diferencias-entre-html-4-y-html-5-significado-de-strict-cu00706b&catid=69:tutorial-basico-programador-web-html-desde-cero&Itemid=192
- [9] Grails is Open Source, Apache 2 License, "**Grails Framework** ", 2005-2016.
<https://grails.org>
- [10] Nacho Brito, "**Manual de Desarrollo Web con GRAILS**", Diciembre 2009, pp 1-208.
- [11] Pivotal Software, Inc. All Rights Reserved. Terms of Use and Privacy, "**SpringSource tool suite**". <http://www.springsource.com/developer/sts>
- [12]] Pivotal Software, Inc. All Rights Reserved. Terms of Use and Privacy, "**Groovy y Grails**"
<http://www.springsource.com/developer/grails>
- [13] Esteban Saavedra, "**Grails Framework para el desarrollo de aplicaciones Web Nº1**", Mayo 2009, pp 1-10. <http://www.slideshare.net/estebansaavedra/grails-framework-para-el-desarrollo-de-aplicaciones-web>

- [14] Esteban Saavedra, **"Grails Framework para el desarrollo de aplicaciones Web N°2"**, Mayo 2009, pp 1-6. <http://www.slideshare.net/estebansaavedra/grails-framework-para-el-desarrollo-de-aplicaciones-web-1394701?related=1>
- [15] Esteban Saavedra, **"Grails Framework para el desarrollo de aplicaciones Web N°3"**, Noviembre 2012, pp 1-9. <http://www.slideshare.net/estebansaavedra/grails-framework-para-el-desarrollo-de-aplicaciones-web-no-3?related=2>
- [16] Esteban Saavedra, **"Grails Framework para el desarrollo de aplicaciones Web N°4"**, Septiembre 2009, pp 1-6. <http://www.slideshare.net/estebansaavedra/grails-framework-para-el-desarrollo-de-aplicaciones-web-no-5?related=3>
- [17] The Groovy project — Groovy is Open Source, Apache 2 License, **"Groovy"**, 2003-2016. <http://www.groovy-lang.org>
- [18] The Groovy project — Groovy is Open Source, Apache 2 License **"The Grails Frameworks Reference Documentation"**, 2003-2016. <http://grails.github.io/grails-doc/latest/>
- [19] The Groovy project — Groovy is Open Source, Apache 2 License, **"Many-to-Many Mapping without Hibernate XML"**, 2003-2016. <https://grails.org/wiki/Many-to-Many%20Mapping%20without%20Hibernate%20XML>
- [20] The Groovy project — Groovy is Open Source, Apache 2 License, **"Plugins Mail"**, Julio 2014. <https://grails.org/plugin/mail>
- [21] The Groovy project — Groovy is Open Source, Apache 2 License, **"Plugins Simple Captcha"**, Septiembre 2014. <https://grails.org/plugin/simple-captcha>
- [22] The Groovy project — Groovy is Open Source, Apache 2 License **"Ajax-Driven Selects in GSP"**, Septiembre 2014. <https://grails.org/wiki/AJAX-Driven%20SELECTs%20in%20GSP>
- [23] JAN, **"Grails and Simple AJAX"**, Abril 2008. <http://jan-so.blogspot.com.es/2008/04/grails-and-simple-ajax.html>
- [24] Edu4Java, **"Tipo de Base de Datos, Cliente SQL y Esquema Video Tutorial 2 SQL en español"**, Noviembre 2010. <https://www.youtube.com/watch?v=CkJsh-5p0OE&feature=relmfu>
- [25] Oracle Corporation and/or its affiliates, **"MySql Workbench"**, 2016. <https://www.mysql.com/products/workbench/>
- [26] Oracle Corporation and/or its affiliates, **"Configuración de Base de Datos en Grails"**, 2016. <https://www.mysql.com/products/workbench/>
- [27] W3Schools.com, **"HTML5 Video"**, 1999-2016. http://www.w3schools.com/html/html5_video.asp
- [28] **"Como insertar un Video con HTML5"**, pp 1-28. <http://anavallasuiza.com/share/presentations/html5e/video.html#slide-9>

- [29] Bruce Lawson y Patrick H. Lauke *"Introduction to HTML5 Video"*, Febrero 2010.
<https://dev.opera.com/articles/introduction-html5-video/>
- [30] Ministerio de Educación Cultura y Deporte, *"HTML5 en la educación"*.
http://www.ite.educacion.es/formacion/materiales/182/cd/cinco/insertar_un_vdeo.html
- [31] Bruce Lawson y Patrick H. Lauke *"Introducción al video HTML5"*, Febrero 2010.
<http://mosaic.uoc.edu/ac/le/es/m8/ud4/>
- [32] Mozilla Developer Network y colaboradores individuales *"Utilizando Audio y Video con HTML5"*, Octubre 2015.
https://developer.mozilla.org/es/docs/Web/HTML/Usando_audio_y_video_con_HTML5
- [33] Microsoft, *"Uso de JavaScript para controlar el reproductor de video HTML5"*, 2016.
<https://msdn.microsoft.com/es-es/library/hh924823%28v=vs.85%29.aspx>
- [34] Alejandro Morales Gámez, Manolo Aguado, Javier Mendoza, etc. miembros de Desarrollo Web *"Manual de JQuery"*, 1999.
<http://www.desarrolloweb.com/manuales/manual-jquery.html>
- [35] W3Schools.com, *"JavaScript"*, 1999-2016. <http://www.w3schools.com/js/>
- [36] W3Schools.com, *"CSS"*, 1999-2016. <http://www.w3schools.com/css/>
- [37] The jQuery Foundation. JQuery License, *"Jquery Ajax"*, 2005-2016.
<http://api.jquery.com/jquery.ajax/>
- [38] The jQuery Foundation. JQuery License, Plugins JQuery UI, *"JQuery-ui"*, 2005-2016.
<http://jqueryui.com>
- [39] The jQuery Foundation. JQuery License, *"Jquery-UI Dialog"*, 2005-2016.
<https://jqueryui.com/dialog/>
- [40] DataTables designed and created by SpryMedia Ltd, *"Plugins JQuery DataTables"*, 2007-2016. <https://www.datatables.net/>
- [41] The GIMP Team, *"GIMP"*, 2001-2015. <http://www.gimp.org/>
- [42] Mozilla Org, *"Video DownloadHelper"*, 1998-2016.
<https://addons.mozilla.org/es/firefox/addon/video-downloadhelper/>
- [43] Mozilla Org, *"Firebug"*, 1998-2016. <https://addons.mozilla.org/es/firefox/addon/firebug/>
- [44] Arsys Internet S.L.U , *"HTML5 vs. Flash"*.
<http://www.arsys.info/general/html5-vs-flash-en-busca-del-nuevo-estandar-interactivo-para-los-desarrollos-web/>
- [45] YouTube, LLC. *"YouTube"*. <https://www.youtube.com/?hl=es&gl=ES>